UNIVERSITY OF CALIFORNIA

Los Angeles

Two Dynamical System Models

Based on Real-World Scenarios:

a Swarming Control Model
and a Surface Tension Model

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Mathematics

by

Wangyi Liu

2011

Report Documentation Page

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **JUN 2011**	2. REPORT TYPE	3. DATES COVERED **00-00-2011 to 00-00-2011**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Two Dynamical System Models Based on Real-World Scenarios: a Swarming Control Model and a Surface Tension Model	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of California, Los Angeles (UCLA),Department of Mathematics,Los Angeles,CA,90095**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Dynamical systems are quite often used to describe complex real-life phenomena. In this dissertation we consider two di erent scenarios where we propose such models. In the rst part we consider the problem of collaborative searching where agents try to search for unknown targets while keeping group formation. This scenario is observed in many animal groups, and can be applied to man- made problems like searching for mines. We use a basic swarming model combined with group decision control for this scenario. We also derive some physical scaling properties of the system and compare the results to the data from the simulations. In the second part we consider a model for the droplet breakup phenomenon. The most important problem of this scenario is how to model surface tension. We explore two di erent models of the di use interface type to describe this scenario namely the Cahn-Hilliard model and the Allen-Cahn model with advection. Using asymptotic methods we correctly predict the breakup condition for the Cahn-Hilliard model. Moreover, we prove that the Allen-Cahn model will not break up under certain circumstances due to a maximum principle. Simuluations in one two, and three dimensions verify the theoretical results and provide more insight into the dynamics.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**107**	

The dissertation of Wangyi Liu is approved.

Jeff D. Eldredge

Lincoln Chayes

Stanley Osher

Andrea L. Bertozzi, Committee Chair

University of California, Los Angeles

2011

To my family,

who told me

to think, and to smile.

TABLE OF CONTENTS

List of Figures

LIST OF TABLES

especially Maggie Albert, Martha Contreras and Babette Dalton, whose work made my graduate study smooth and comfortable.

I would like to thank all my friends that I meet online or offline, especially those who took me to various events. They colored my life at UCLA into a wonderful experience.

Finally I would like to thank my parents and my grandparents for all their support since I was born. They are role models for me in every aspect of life, and made me what I am today.

This dissertation uses figures from [DCB06, VCB95, SSO94, YFL05, TSO92, KKL04], and I would like to thank American Physical Society, Elvesier, and Cambridge University Press for their permission to reproduce the figures.

Vita

1989   Born, Beijing, China.

2007   B.S. (Mathematics), Peking University

PUBLICATIONS

Wangyi Liu, Martin B. Short, Yasser E. Taima, and Andrea L. Bertozzi, Multiscale Collaborative Searching Through Swarming, Proceedings of the 7th International Conference on Informatics in Control, Automation, and Robotics (ICINCO), Portugal, June 2010.

ABSTRACT OF THE DISSERTATION

Two Dynamical System Models

Based on Real-World Scenarios:

a Swarming Control Model
and a Surface Tension Model

by

Wangyi Liu

Doctor of Philosophy in Mathematics

University of California, Los Angeles, 2011

Professor Andrea L. Bertozzi, Chair

Dynamical systems are quite often used to describe complex real-life phenomena. In this dissertation we consider two different scenarios where we propose such models. In the first part we consider the problem of collaborative searching, where agents try to search for unknown targets while keeping group formation. This scenario is observed in many animal groups, and can be applied to man-made problems like searching for mines. We use a basic swarming model combined with group decision control for this scenario. We also derive some physical scaling properties of the system and compare the results to the data from the simulations. In the second part we consider a model for the droplet breakup phenomenon. The most important problem of this scenario is how to model surface tension. We explore two different models of the diffuse interface type to describe this scenario, namely the Cahn-Hilliard model and the Allen-Cahn model with advection. Using asymptotic methods we correctly predict the breakup condition for the Cahn-

Hilliard model. Moreover, we prove that the Allen-Cahn model will not break up under certain circumstances due to a maximum principle. Simuluations in one, two, and three dimensions verify the theoretical results and provide more insight into the dynamics.

Part I

Collaboration Searching Using Swarming Technique

CHAPTER 1

Introduction to the Collaborative Searching Problem

Collaborative control has long attracted research interest and different scenarios have been studied. A variety of cooperative control model that lead to agent swarming, which makes the agents to form a global pattern under local interactions. These kinds of model are originally designed to model the behavior of animal groups. Other models relies on specific tasks that the agents need to conduct, thus requires other tools like the consensus problem or coverage control.

Among swarming models, two distinct types are first-order and second-order models. First-order models directly control the velocity of the agents, while second-order models only control the acceleration of the agents. The famous Vicsek model [VCB95] is a good example of a first-order model. The agent movement pattern is given as

$$x_i(t+1) - x_i(t) = v_i(t)\Delta t, \tag{1.1}$$

Here x_i denotes the position of the i-th agent, and v_i denotes the velocity of the agent. The velocity $v_i(t)$ has a constant length of v and a direction angle of $\theta_i(t)$, which is determined by

$$\theta_i(t+1) = <\theta(t)>_r + \Delta\theta. \tag{1.2}$$

Here $<\theta(t)>_r$ denotes the average angle for all agents within radius r of the

agent i and $\Delta\theta$ is a random noise of given strength. Depending on the density of agents and strength of noise, different pattern may occur. When noise is large, the agents move randomly. When the noise is small and density is high, the agents become all ordered to one direction. When the noise is small and density is also small, the agents will form groups that move together. See Fig. 1.1.
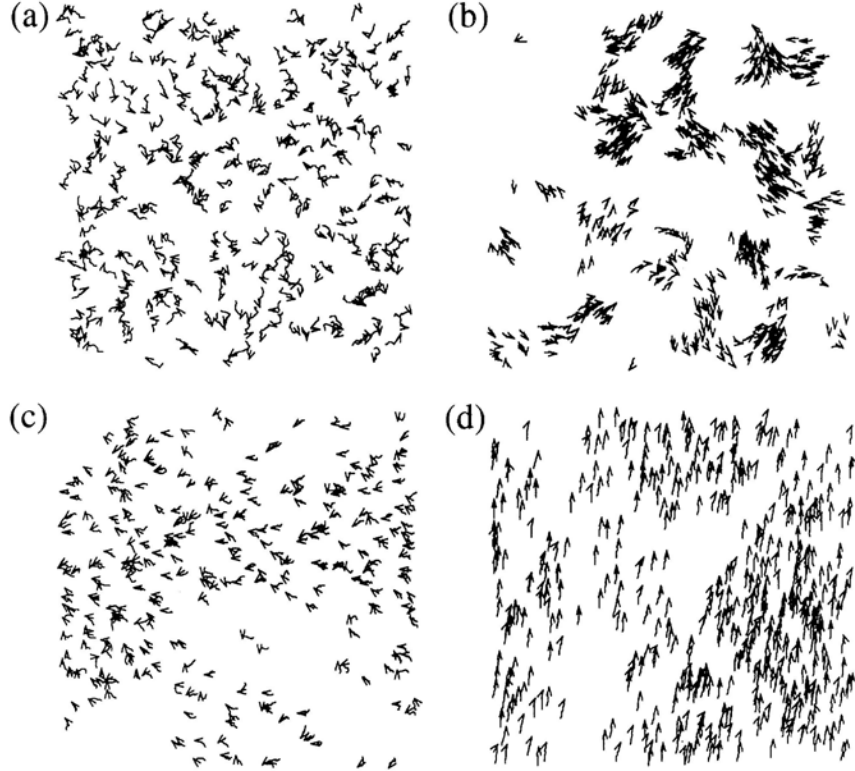
Figure 1.1: Numerical simulation for Vicsek model, from [VCB95], copyright 2006 The American Physical Society, reproduced with permission. A and C are when noise is large, where A is the initial value and C is after some time has passed. B is the case when noise is small and density is also small, and D is the case when noise is small but density is large.

An example of a second-order swarming model is described in [DCB06].

$$\frac{dx_i}{dt} = v_i \ ,$$
(1.3)

and

$$m_i \frac{dv_i}{dt} = (\alpha - \beta |v_i|^2)v_i - \nabla U(x_i),$$
(1.4)

where

$$U(x_i) = \sum_{j=1}^{N} C_r e^{-|x_i - x_j|/l_r} - C_a e^{-|x_i - x_j|/l_a} \ .$$
(1.5)

In this system, each agent of the swarm is subject to self-propulsion, drag, and attractive, repulsive, and velocity alignment forces from each of the other agents. Under different parameters, the agents may form clumps, ring clumping, or rings. See Fig. 1.2. For more examples of swarming models, see [JK04], [VCB95], and [SPL08].

Models that require agents to perform given tasks differ from each other due to the vast diversity of scenarios. Here we give a few examples. The coverage control problem [BC05] requires the agents to perform spatially-distributed sensing tasks. It then assign different regions to different agents to maximize the efficiency. The pursuit-evasion scenario [BBH09] considers the case where a group of predators hunting a single prey. It then designs the best movement strategy for the predator. The consensus problem [GCB08] considers how to determine the true value when multiple agents take different measurements of the value. It is especially hard when the communication is limited. The team-forming problem [SB09] assumes that the agents cannot finish a task alone. Different tasks require different combination of agents, and tasks are assigned dynamically over a large region. It then computes the agent movement dynamically when task is assigned. Using such collaborating sensors to detect and locate targets within an area has been studied in reference to the "mine counter-measure" problem [CMT03], the specific military task of locating ground or water-based mines.
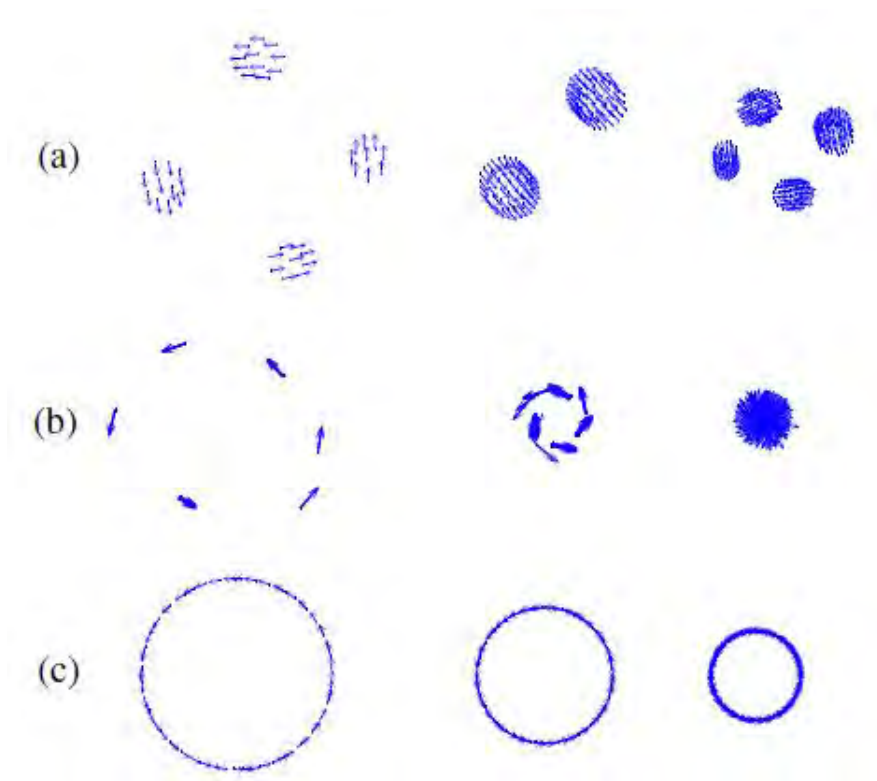
Figure 1.2: Numerical simulation for the model from [DCB06], copyright 2006 The American Physical Society, reproduced with permission. A, B and C are clumps, ring clumping, and rings respectively.

In this part, we develop a multiscale search and target-locating algorithm for a type of mine counter-measure problem in which a number of independent agents are given the task of determining the precise location of targets within a domain. The algorithm is designed to handle problems where the scale of the target sensing radius is much smaller than the domain size. The focus of this work is to identify optimality of the algorithm as a function of the swarm size, the number of agents per group and the distribution of resources into different groups.

We assume a simply-connected domain, and use noisy sensors that detect a scalar quantity emitted by each target, but only when an agent is within a fixed distance r_s from a target. We control the motion of the agents with a model that makes the individuals form distinct swarms, and present filtering techniques that allow for locating targets despite noisy data. The inspiration for this approach comes in part from biology, as in the example of birds forming flocks when flying and searching for food [Tra07].

Next, we analytically derive some of the system's main scaling properties, such as the relationship between swarm size, distance between agents and target sensing radius, and compare with the experimentally recorded data. We conclude that our analytical approach matches well the data from the simulations.

We assume a sensing radius r_s much smaller than the domain but comparable to or less than the swarm size. Other assumptions, however, may require different algorithms. For example, in [BLT09] the sensing radius is infinite, but sensing is limited by obstacles, and in [Olf07] communication between agents is not always possible.

We consider one such scenario and its corresponding algorithm in the next chapter. We then consider some properties of the algorithm in Chapter 3 , fol-

lowed by a conclusion and ideas for future research in Chapter 3.3.

CHAPTER 2

The Collaborative Searching Model

We consider M targets in a two-dimensional simply-connected domain, that is, a flat enclosed area with no holes, and N agents able to move freely within the domain. Each target emits a radially symmetric scalar signal $g(r)$ that decays with the distance r from the target, and drops to zero at some r_s, the target's sensing radius. Agents detect this signal with an additional Gaussian, scalar white noise component added. If an agent is within the sensing radius of multiple targets, it detects only the sum of the individual signals, again with a noise component added. We suppose that an agent takes sensor readings at regular intervals (once per "time step") spaced such that the noise between time steps can be assumed independent.

The algorithm accomplishes 3 tasks: it filters the noisy sensor data, controls the coordinated movement of the agents based on this data, and determines when a target has been acquired and where it is located. Section 2.1 focuses on the techniques we use to process sensor data, Section 2.2 describes the movement control of the agents, and Section 2.3 describes the method for locating a target.

2.1 Sensor Data Processing

Due to noise in the agent sensor readings and the sensing radius r_s being finite, we employ two distinct filters to the data from the readings: a Kalman filter and

a Cumulative Sum (CUSUM) filter. The Kalman filter reduces or eliminates the noise in the data, while the CUSUM filter is well-suited to determining whether or not an agent is within the sensing radius of a target.

The sensor data model follows as a mathematical formula. As explained in Section 1, the formula describes sensor readings as the sum of scalar signals that depend only on the distance from a target, together with a noise component.

Given the M targets at positions y_j and an agent i with current position $x_i(t_k)$ at timestep k, the agent sensor reading $s_i(t_k)$ is given by

$$s_i(t_k) = \sum_{j=1}^{M} g(|y_j - x_i(t_k)|) + n_i(t_k) \, , \tag{2.1}$$

where $n_i(t_k)$ denotes sensor noise and $g(r)$ is the signal strength at a distance r from the target. For simplicity, we assume $g(r)$ is isotropic, smooth, decaying, the same for all targets, and has a cutoff at r_s.

2.1.1 Kalman Filtering

Before using the agent sensor readings to locate targets or control agent motion, we pass the sensor readings through a Kalman filter. Since the signal from the target is presumed to be varying smoothly with the distance to the target r (up to the cutoff point r_s) as the agents navigate the environment, a Kalman filter is a natural choice to eliminate or reduce noise in the sensor readings. The Kalman filter takes the sensor reading $s_i(t_k)$ of agent i at time t_k, and converts it into the filtered data $f_i(t_k)$ according to

$$P_i(t_k) = \frac{P_i(t_{k-1})R_i(t_k)}{P_i(t_{k-1}) + R_i(t_k)} + Q_i(t_k) \, , \tag{2.2}$$

and

$$f_i(t_k) = f_i(t_{k-1}) + \frac{P_i(t_k)(s_i(t_k) - f_i(t_{k-1}))}{P_i(t_k) + R_i(t_k)} \, . \tag{2.3}$$

Here $R_i(t_k)$ is the square of the noise amplitude, known or estimated by the agent, and $Q_i(t_k)$ is the square of the change of the signal amplitude between two time steps, either fixed beforehand or estimated using the current velocity of the agent (in this chapter, it is fixed beforehand). $P_i(t_k)$ is roughly the variance of the sensor reading's amplitude. The output $f_i(t_k)$ of this filter is then used in target locating, as described in Section 2.3.

2.1.2 Threshold Check and the CUSUM Filter

Before attempting to locate targets, an agent needs to determine whether or not it is receiving an actual signal, rather than just noise. In other words, an agent needs to determine whether it is within the sensing radius of a target at each time-step t_k. This information is then used both in controlling the movement of the agents and in determining when to begin estimating a target's position. In order to determine the sensing status of an individual agent, we employ a CUSUM filter, as this type of filter is well-suited to determining abrupt changes of state [PAG54], and has been used in the similar task of boundary tracking [JB07, CWT09]. The filter keeps a sort of running average of the signal and notes when this average seems to have risen above a cetain threshold, indicating that the agent is now within the sensing radius of a target. As the noise is effectively summed up by the filter, it tends to cancel out.

In the original form of the CUSUM filter, we imagine a sensor that returns a sequence of independent observations $s(t_1)...s(t_n)$, each of which follows one of two probability density functions: a pre-change function g_0 and a post-change function g_1. The log-likelihood ratio is

$$Z(t_k) = \log[g_1(s(t_k))/g_0(s(t_k))] \ , \tag{2.4}$$

and we define the CUSUM statistic as

$$U(t_k) = \max(0, Z(t_k) + U(t_{k-1})), \quad U(t_0) = 0 . \tag{2.5}$$

We then choose a threshold $\bar{U}$, and when $U(t_k) \geq \bar{U}$ for the first time, the algorithm ends and we declare that the state has changed from g_0 to g_1. The threshold should be chosen so as to minimize both false-alarms (these happen more frequently for small $\bar{U}$) and time to detection (this gets larger as $\bar{U}$ increases).

In our system, we choose the special case where sensor reading follows a Gaussian distribution. In the pre-change state g_0, the agent is outside the sensing radius of any target and reads only noise, which we model as a Gaussian with zero mean and variance σ^2. In the post-change state g_1, the agent enters the sensing radius of a target, and although the probability distribution is still a Gaussian with the same variance, the mean is now larger than zero, which we set to be $2B$. Then

$$
\begin{aligned}
Z(t_k) &= \log \left[\frac{e^{-[s(t_k)-2B]^2/2\sigma^2}/(\sigma\sqrt{2\pi})}{e^{-s(t_k)^2/2\sigma^2}/(\sigma\sqrt{2\pi})} \right] \\
&= \frac{-[s(t_k) - 2B]^2}{2\sigma^2} + \frac{s(t_k)^2}{2\sigma^2} \\
&= \frac{2B}{\sigma^2}[s(t_k) - B] .
\end{aligned}
\tag{2.6}
$$

We also modify the algorithm so that it can detect status changes both into and out of detection zones. Thus, we implement two filter values: $U_i(t_k)$ to determine when an agent has entered a zone, and $L_i(t_k)$ to determine if they have left a zone. We also define a binary function $b_i(t_k)$ which denotes the status of an agent; $b_i(t_k) = 1$ denotes that the agent is near a target and $b_i(t_k) = 0$ means otherwise. The filter values all start at zero, and are updated according to

$$U_i(t_k) = \max(0, s_i(t_k) - B + U_i(t_{k-1})) , \tag{2.7}$$

$$L_i(t_k) = \min(0, s_i(t_k) - B + L_i(t_{k-1})) \ , \qquad (2.8)$$

and

$$b_i(t_k) = \begin{cases} 1 & b_i(t_{k-1}) = 0, \ U_i(t_k) > \bar{U} \\ 0 & b_i(t_{k-1}) = 1, \ L_i(t_k) < \bar{L} \qquad (2.9) \\ b_i(t_{k-1}) & \text{otherwise.} \end{cases}$$

In addition, when the status of agent i changes, we reset the corresponding U_i or L_i to zero. Here we have set the constant coefficient $\frac{2B}{\sigma^2}$ to 1, for convenience.

Recall that B is a sensor value that is less than the predicted mean when inside a sensing radius, and $\bar{U}$ is our chosen detection threshold. So, when the agent is near a target, the sensor reading $s_i(t_k)$ tends to be larger than B, causing $U_i(t_k)$ to grow quickly until it is larger than $\bar{U}$, indicating a change in status. The converse is true if an agent leaves the sensing region of a target. The values of the various parameters of the filter are problem-specific, and should be estimated in a manner that minimizes both false-alarms while keeping the average time to detection as low as possible, as mentioned above.

An example of sensor reading from an agent within our current simulations can be seen in Fig. 2.1. The Kalman filter does a good job of reducing noise, bringing the sensor readings much closer to the true signal. Near the middle of the plot, the agent enters into the sensing radius of a target, and this is reflected by a transition within the CUSUM filter from $b = 0$ to $b = 1$. There is, as expected from the behavior of CUSUM, a slight delay between when the agent actually enters into the radius and when this transition of b occurs. After spending some time within the sensing radius, the estimated target location stabilizes, the agent subtracts the true signal from its measurements (this will be explained in Section 2.3), and the agent leaves to find further targets.
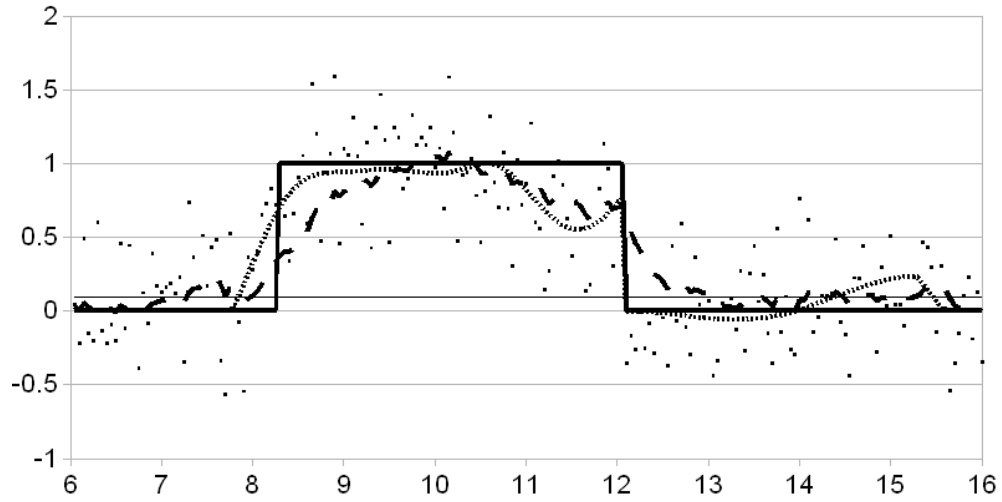
Figure 2.1: Example filter output from one agent as a function of time, from one of our simulations. The densely-dotted line represents the signal before noise that should be detected by the agent. The dots are the actual noisy signal detected by the agent at each time step (i.e., the densely-dotted curve plus noise). The thicker step function is the signal status returned by the CUSUM filter, and the thinner straight line represents the value $B = 0.1$. The sparsely-dashed curve is the output of the Kalman filter when applied to the detected noisy signal.

2.2 Agent Movement Control

We have chosen to control the movement of our agents by breaking up our total agent population N into a number of distinct, leaderless "swarms". This is done for a variety of reasons. Firstly, it increases robustness, as any individual swarm member is not critical to the functioning of the swarm as a whole. Secondly, since we imagine that any sensor data acquired by readings from these agents is local in space, a swarm provides a method of extending the effective sensing zone to the whole swarm. Thirdly, a swarm of nearby agents may use their combined measurements to decrease sensor noise. Finally, the swarm provides the ability to locate targets via triangulation or gradient methods. Each of the various swarms may search within a different region of space if a divide-and-conquer tactic is desired, or each swarm may be free to roam over the entire region. In the following two sections we mainly focus on the control of one swarm.

Since the agents have a limited sensing radius, we choose to employ two different phases of swarm motion. When there are no targets nearby, the agents should move through the space as quickly and efficiently as possible, performing a simple flocking movement as legs of a random search. After a signal is sensed via the CUSUM filter, the agents should stop, then slowly move around the area, searching for the exact position of the nearby target. We call these two phases the searching phase and the target-locating phase, respectively. For a general idea of the two types of motion, see Fig. 2.2.

2.2.1 The Swarming Model

We choose a second-order control algorithm similar to that described in [DCB06] and [CHD07], which has been successfully implemented as a control algorithm

14
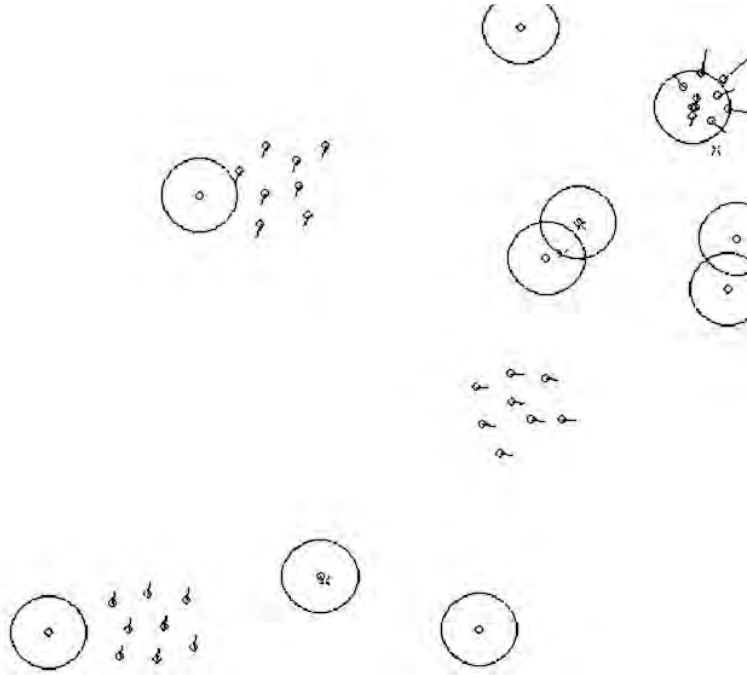
Figure 2.2: A screenshot from the simulation. Four swarms with eight agents each are used. Three of them are in the searching phase, and the upper right swarm is in the target-locating phase. Large circles around targets denote the sensing radius. Small crosses are already registered targets.

for second-order vehicles on real testbeds [NCT05, LHH07]. In addition to the self-propulsion, drag, and attractive, and repulsive forces that are described in the original model[DCB06], the agent is also subject to a velocity alignment force. The position x_i and velocity v_i of an individual agent i with mass m_i in a swarm of N agents are governed by

$$\frac{dx_i}{dt} = v_i \ , \tag{2.10}$$

and

$$m_i \frac{dv_i}{dt} \quad = \quad (\alpha \ - \ \beta|v_i|^2)v_i \ - \ \nabla U(x_i) \ + \ \sum_{j=1}^{N} C_o(v_j - v_i) \quad , \tag{2.11}$$

where

$$U(x_i) = \sum_{j=1}^{N} C_r e^{-|x_i - x_j|/l_r} - C_a e^{-|x_i - x_j|/l_a} \ . \tag{2.12}$$

C_r and l_r are the strength and characteristic length of the repulsive force, respectively, and C_a and l_a are the corresponding values for the attractive force. C_o is the velocity alignment coefficient and α and β are for self-propulsion and drag, respectively. Depending on these parameters, the swarms can undergo several complex motions [DCB06], two of which are flocking and milling, and in some cases the swarms can alter motion spontaneously [KMK07]. For our purposes, we simply set these parameters to obtain the type of motion desired.

2.2.2   Searching Phase

In this phase, the agents move together in one direction as a uniformly-spaced group travelling with a fixed velocity. Since the agents know nothing about the location of targets, a random search is chosen here. Specifically, we use a Lévy flight, which is optimally efficient under random search conditions [VBH99], and is the same movement that some birds employ [Tra07]. To accomplish this type of search, we simply command the swarm to turn by some random angle after

flocking for some random amount of time. For a Lévy flight, the time interval Δt between two turns follows the heavy-tailed distribution

$$P(\Delta t) \sim \Delta t^{-\mu} \ , \tag{2.13}$$

where μ is a number satisfying $1 < \mu < 3$. The value of μ should be chosen optimally according to the scenario in question, as in [VBH99]. For destructive searching (where targets, once located, are no longer considered valid targets), μ should be as close to 1 as possible. For non-destructive searching (where located targets remain as valid future targets), the optimal $\mu \sim 2 - 1/[\ln(\lambda/r_s)]^2$, where λ is the mean distance between targets and r_s is the sensing radius.

2.2.3 Target Locating Phase

When enough agents agree that a target is nearby (see Section 2.1.2), the target-locating phase begins. This minimum number is set by the swarm consensus parameter p, such that the swarm decides to enter this phase when $p\%$ of the agents or more in the swarm are sensing a target. Once in this phase, we want the agents to move only towards the target, so we remove the velocity alignment force ($C_o = 0$), disable self-propulsion ($\alpha = 0$), and issue a halt command so that all agents begin target locating with zero velocity. In addition, data from agents within the sensing radius is used to continually estimate the position $\bar{y}$ of the target (see section 2.3), and the agents in the swarm then try to move towards it, thus attracting other agents in the swarm not yet in the sensing radius to move closer to the target as well. To make the agents move towards the target, we add another potential in Eq. 2.12,

$$U_c = C_c(x_i - \bar{y})^2/2 \ , \tag{2.14}$$

where $\bar{y}$ is the estimated position of the target, and C_c is an adjustible parameter. The full control equations in the target-locating phase therefore become Eq. 2.10 and

$$m_i \frac{dv_i}{dt} = -\beta |v_i|^2 v_i - \nabla U(x_i) \,, \tag{2.15}$$

where

$$U(x_i) \quad = \quad \frac{1}{2} C_c (x_i \ - \ \bar{y})^2 \ + \ \sum_{j=1}^{N} C_r e^{-|x_i - x_j|/l_r} - C_a e^{-|x_i - x_j|/l_a} \quad . \tag{2.16}$$

To show that this system converges to a stationary swarm centered on the target, we note that the total energy of the target locating system,

$$E = \frac{1}{2} \sum_{i=1}^{N} m_i |v_i|^2 + \sum_{i=1}^{N} U(x_i) \,, \tag{2.17}$$

serves as a Lyapunov function, so that the collective tends to minimize it. That is,

$$\dot{E} = -\beta \sum_{i=1}^{N} |v_i|^4 \leq 0 \,. \tag{2.18}$$

Hence, velocities will eventually reach zero (due to drag) and the swarm members will spatially re-order themselves so as to minimize the potential energy, forming a regular pattern centered at the target position. This stationary state serves as a spiral sink, however, so the swarm tends to oscillate about the target position for some amount of time that depends on the value of C_c, with a high C_c yielding less oscillation. However, since the potential being minimized now includes a term that is effectively attracting all of the agents towards the center of mass, the swarm will be more compact than it was before the target locating potential was added, so too large of a C_c will make the swarm smaller than desired. In practice, we want C_c just large enough to minimize the oscillations in space without making the swarm get too compact.

2.3 Locating Targets

During the target-locating phase of motion, all agents of the swarm that are within the sensing radius keep a common register of all of their positions and signal readings made since entering the radius (see "Threshold Check", Section 2.1.2, above). The agents then use a least-squares algorithm to give an estimate $\bar{y}$ of where the target is located via

$$\bar{y} = \min_y \sum_{k=1}^{N'} \left[g(|y - x(t_k)|) - f(t_k) \right]^2 \ , \tag{2.19}$$

where N' is the number of sensor readings in the common register.

Solving this least-squares minimization is straightforward, but certain assumptions for workability and precision are needed. It is assumed that the form of $g(r)$ is known by the agents for the algorithm to work. For certain classes of targets and scalar fields, we believe this assumption is fair. To precisely estimate a target's location, we also assume that only one target is within sensing range, or that target sensing radii do not overlap significantly, so one target is much closer to the agents than any other target. When the sensing radius is small compared to the average distance between targets, these assumptions should hold true. If, instead they prove to be invalid for the particular system at hand, other methods such as gradient estimation could be used.

If the estimated position of the target stabilizes, it is considered to have been located, and the agents register the position of the target and return to the searching phase. The model signal $g(r)$ from the registered target will be subtracted from further sensor readings so that it is not detected again, a form of destructive searching. We thus modify Eq. 2.1 to read:

$$s_i(t_k) \ = \ \sum_{j=1}^{M} g(|y_j - x_i(t_k)|) \ + \ n_i(t_k) \ - \ \sum_{j=1}^{M'} g(|\bar{y}_j - x_i(t_k)|) \ , \tag{2.20}$$

19
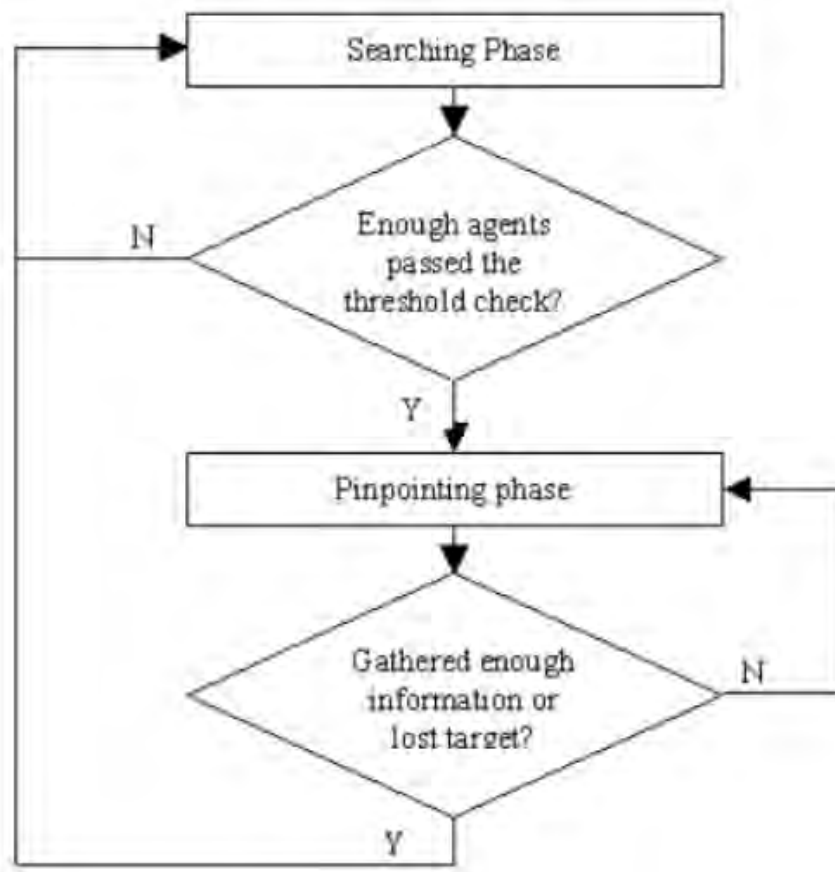
Figure 2.3: A simple flowchart of the algorithm.

where M' is the total number of registered targets. Note that the positions of these targets may or may not be accurate, due to noise and other errors. If, instead of the estimated target location stabilizing, the agents lose track of the target, they simply return to the searching phase without registering the target. For a general idea of the entire algorithm, see Fig. 2.3.

CHAPTER 3

Property and Evaluation of the Algorithm

3.1   Numerical Performance Evaluation

Two main criteria for the evaluation of this algorithm are efficiency and accuracy. These are roughly determined by the two phases: efficiency is mainly related to the swarming phase, while accuracy is mainly related to the target-locating phase. To evaluate the performance of the algorithm, we divided the agents into groups, ran the algorithm, and took the following measurements: the average time needed for a swarm to locate one target (average time), the average distance between the actual and estimated target positions (average error), and the percentage of registered positions that are not within any actual sensing radius (percentage false registers). Note that the false registers are not included in the average error calculation.

We ran computer simulations of the algorithm in a dimensionless 20 by 20 area, with a total of 32 agents and a dimensionless sensing radius of 1. The signals had a Gaussian form, with a peak signal-to-noise ratio of about 10.5 dB. Two cases were considered. In the first case, there were 20 targets and we restricted the duration of the simulation, the main goal being that of measuring efficiency. In the second case, we distributed just 5 targets randomly, and used a much longer time limit, with the main goal of measuring accuracy. In either case, the simulation ends either when time runs out or when all targets are found. For each

case, we performed 100 trials and calculated the average of the measurements.

Since we considered multiple groups of agents, it was important to decide how they must cooperate with one another. We tried two different policies. One was a simple divide-and-conquer tactic where we divide the whole region into sub-regions [EFS05, HCH06] before the simulation, with each swarm in charge of a single sub-region, remaining within that area the entire time, and performing a Lévy flight search pattern confined to its designated area. The other policy allows the swarms to search the entire region independently of one another. In the results, we denoted the use of the divide-and-conquer tactic with an asterisk (*).

An important factor that influenced our results is the number of groups into which we divided the agents, or equivalently N, the number of agents in each swarm. We therefore present the results for several choices of N. They are in Tables 3.1 and 3.2, with the associated plots presented in Figs. 3.1 and 3.2.

From the tables and their corresponding plots we see that the number of agents in the swarm works as a balance between accuracy and efficiency. As could have been anticipated, larger swarms give more accurate results (smaller target position error, less false registers), while multiple, smaller swarms make the search more efficient (faster detection time). To have an acceptably low error and low false target registration rate, groups of at least four agents should be used. This is perhaps due to the fact that at least three agents are needed to locate a target, using triangulation. Also, we note that the divide-and-conquer tactic seems to work somewhat better for this scenario.

Table 3.1: Case 1: 20 targets, time limit 50.0. Asterisks denote the use of the divide-and-conquer tactic.

Swarms	Agents/ swarm	Average time	Average error	False reg.
1	32	9.17	0.163	9.77%
2*	16	4.83	0.155	8.40%
2	16	5.45	0.159	11.90%
4*	8	3.15	0.158	8.68%
4	8	3.52	0.16	10.59%
8*	4	2.67	0.208	9.91%
8	4	2.9	0.200	11.73%
16*	2	2.64	0.257	15.59%
16	2	2.64	0.253	15.17%
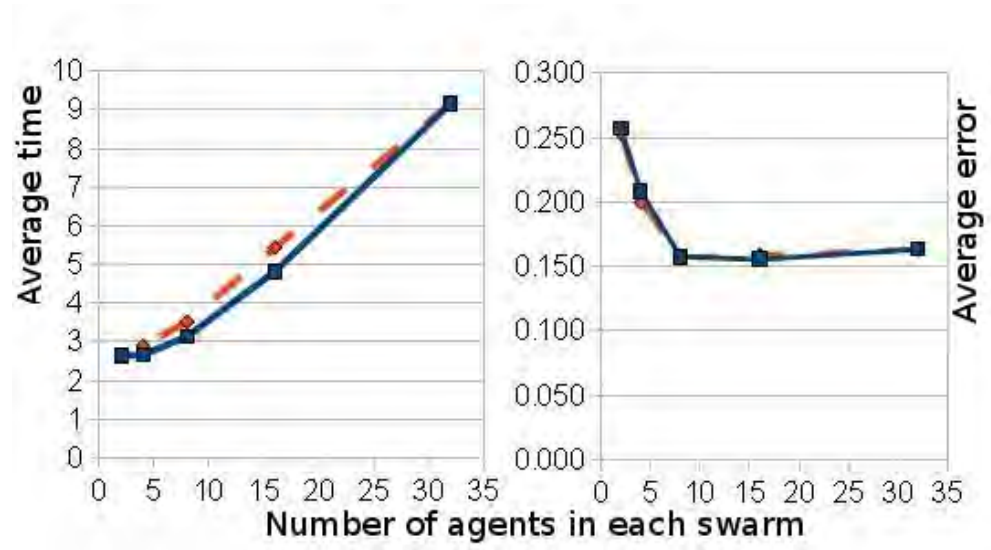
Figure 3.1: Average search time (left) and average error (right) as a function of the number of agents in each swarm for case 1 (20 targets and time limit 50.0). The continuous line is for the divide-and-conquer tactic and the dashed line is for the results without divide-and-conquer.

Table 3.2: Case 2: 5 targets, time limit 200.0. Asterisks denote the use of the divide-and-conquer tactic.

Swarms	Agents/ swarm	Average time	Average error	False reg.
1	32	45.53	0.128	10.76%
2*	16	25.51	0.116	8.06%
2	16	26.89	0.117	8.95%
4*	8	14.22	0.134	8.96%
4	8	16.64	0.118	8.79%
8*	4	8.35	0.161	7.24%
8	4	10.58	0.172	8.97%
16*	2	8.31	0.223	11.97%
16	2	8.91	0.252	13.79%
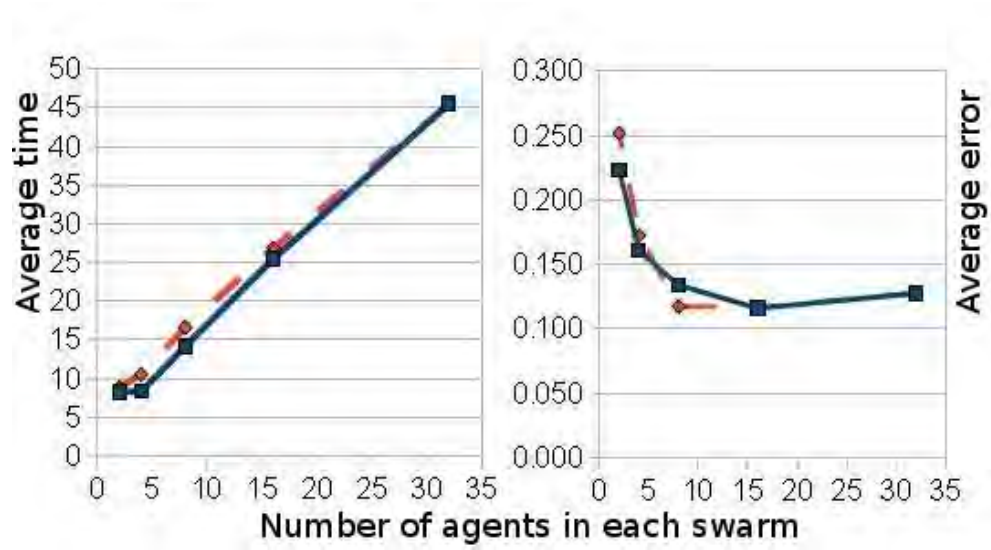
Figure 3.2: Average search time (left) and average error (right) as a function of the number of agents in each swarm for case 2 (5 targets and time limit 200.0). The continuous line is for the divide-and-conquer tactic, and the dashed line is the for the results without divide-and-conquer.

26

3.2 Scaling Properties

Having noted the results above, one may wonder how these are affected by the various scales present in the system, such as the swarm size, distance between agents, target sensing radius, etc. Below we present some arguments for determining optimal search parameters given these scales.

3.2.1 Estimating the Swarm Diameter

We first define a measure for the swarm size, the swarm diameter $D = \max(|x_i - x_j|)_{i=1}^N$, where N is the number of agents in the swarm. Let us also define the inter-agent distance $l = |x_i - x_j|$ for any two nearest-neighbor agents i, j.

For the remainder of this section (and for the results in Fig. 3.4), we choose the parameters of motion so that the system is either in regime VI (catastrophic) or VII (H-stable) as defined in [DCB06], with the swarms flocking naturally in VII, and in VI due to the velocity alignment term C_o in Eq. 2.11. Under these regimes, D and l stabilize after a transient period, so for the purposes of this section we will consider them to be constant in time. In such a stable swarm, agents are uniformly distributed in space along a hexagonal pattern, so that the swarm diameter D and inter-agent length l are related geometrically as follows: since the area occupied by a single agent in the swarm is $A_a \approx \pi l^2/4$ and the total swarm area is $A_s = \pi D^2/4 \approx N A_a$, then

$$D \simeq \sqrt{N} l \ .\tag{3.1}$$

Thus, D scales with l, and for $N = 16$ (as used in Fig. 3.4) we get $D \simeq 4l$. Since l is approximately the distance that minimizes the inter-agent potential of Eq. 2.12, we can easily adjust the swarm diameter D by varying the system parameters.
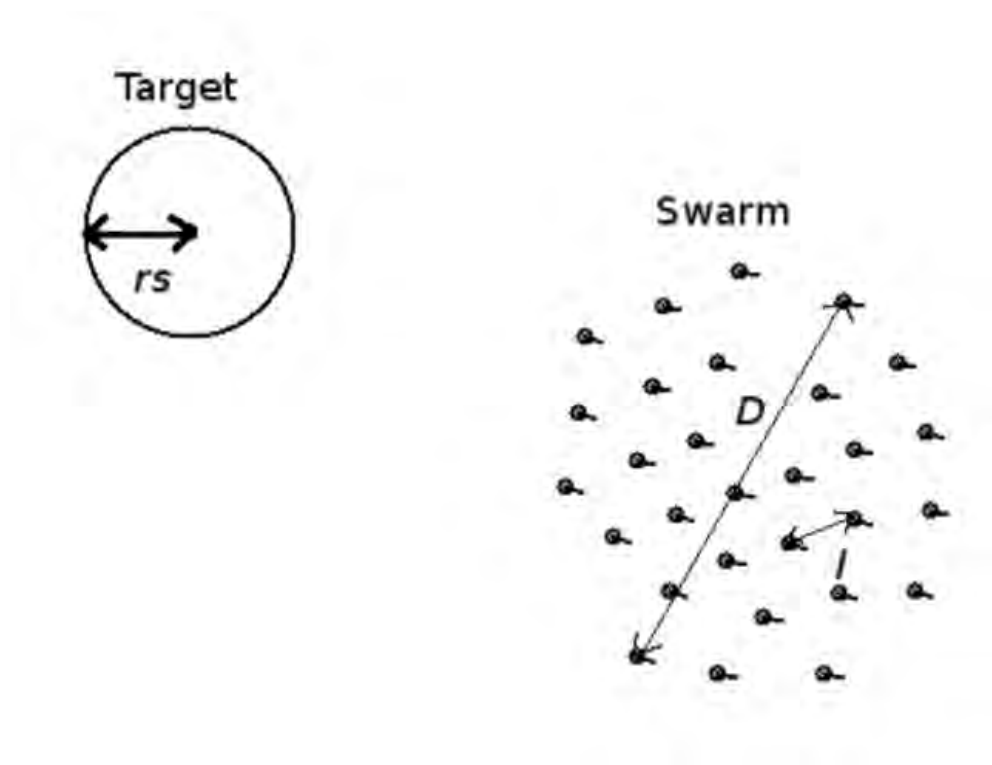
Figure 3.3: Scales influencing target locating time: the swarm diameter D, inter-agent length l, and sensing radius r_s.

3.2.2 An Upper Bound on the Optimal Swarm Diameter

Consider a setting with one target of sensing radius r_s and one swarm of diameter D, as in Fig. 3.3. We measure the average time to locate the target $\bar{T}$ by starting the swarm at the center of the search field at t_0, placing the target at a random location within the field, allowing the simulation to run until time T when the target is found, and averaging these T values over many runs of the simulation. As D grows, we observe (see Fig. 3.4) that $\bar{T}$ decreases until an optimal swarm diameter D_{opt} is reached, after which $\bar{T}$ increases again, growing without bound. We wish to explain this, by first finding an upper bound on D_{opt}.

Let us begin by fixing the swarm consensus percentage at $p = 25\%$; i.e., the swarm of agents decides a target is present when $1/4$ of the agents or more are within the sensing area of a target, $A_t = \pi r_s^2$ (see Fig. 3.5). Clearly, the borderline case between detection and non-detection occurs when the target area is completely subsumed within the swarm area, yet there are only just enough agents (25% of the total) within the target sensing radius to detect it. If we assume a constant density of agents in the swarm, this means that we have detection when the target area is at least $p\% = 1/4$ of the swarm area. So, it must be the case that

$$D \leq 4r_s \, , \tag{3.2}$$

or else the target will not be detected at all. This condition therefore gives an upper bound for D_{opt}. Condition (3.2) can also be written in terms of l, in which case $l \leq r_s$; the borderline case is illustrated in Fig. 3.5(b). In Fig. 3.6, snapshots from a simulation show how the swarm flies over the target without being able to detect it in a case when condition (3.2) is violated.
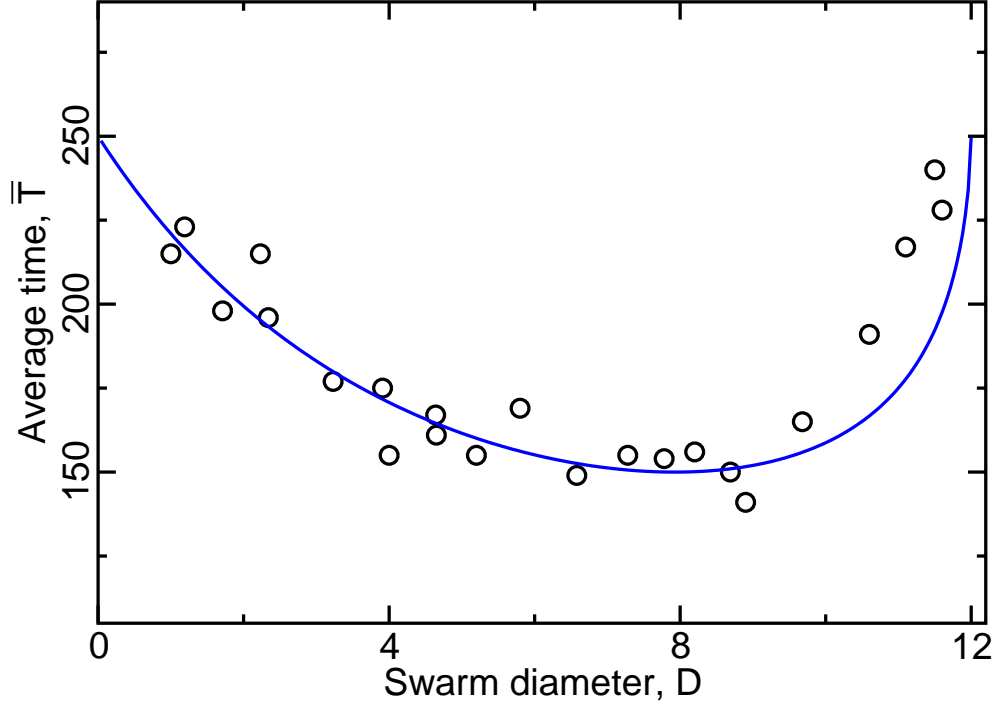
Figure 3.4: Average time to reach a target $\bar{T}$ as a function of the swarm diameter D for $r_s = 3.0$. Values of D were obtained over a range $C_r = 2.0 - 12.0$, $l_r = 0.2 - 0.7$, with $C_a = l_a = 1.0$. Averaging was carried out over 200 simulated trials, yielding the numerical results (circles); the theoretical results of Eq. 3.6 with the best fitting τ are shown as a line. The number of agents $N = 16$, and $D_{\mathrm{opt}} \approx 8$.
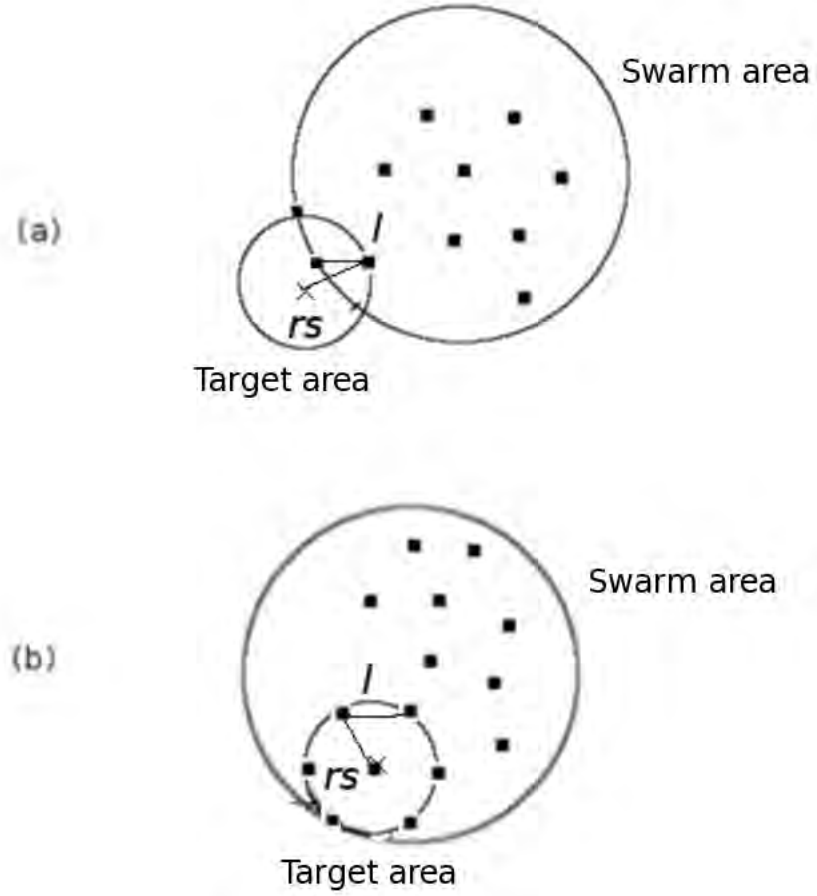
Figure 3.5: Sensing configurations for the case when 4 or more agents are required to sense the target before it can be detected. (a) Though there is overlap between the swarm and target, too few agents can sense the target for it to be detected. (b) The largest inter-agent distance l while still allowing for detection, $l = r_s$.
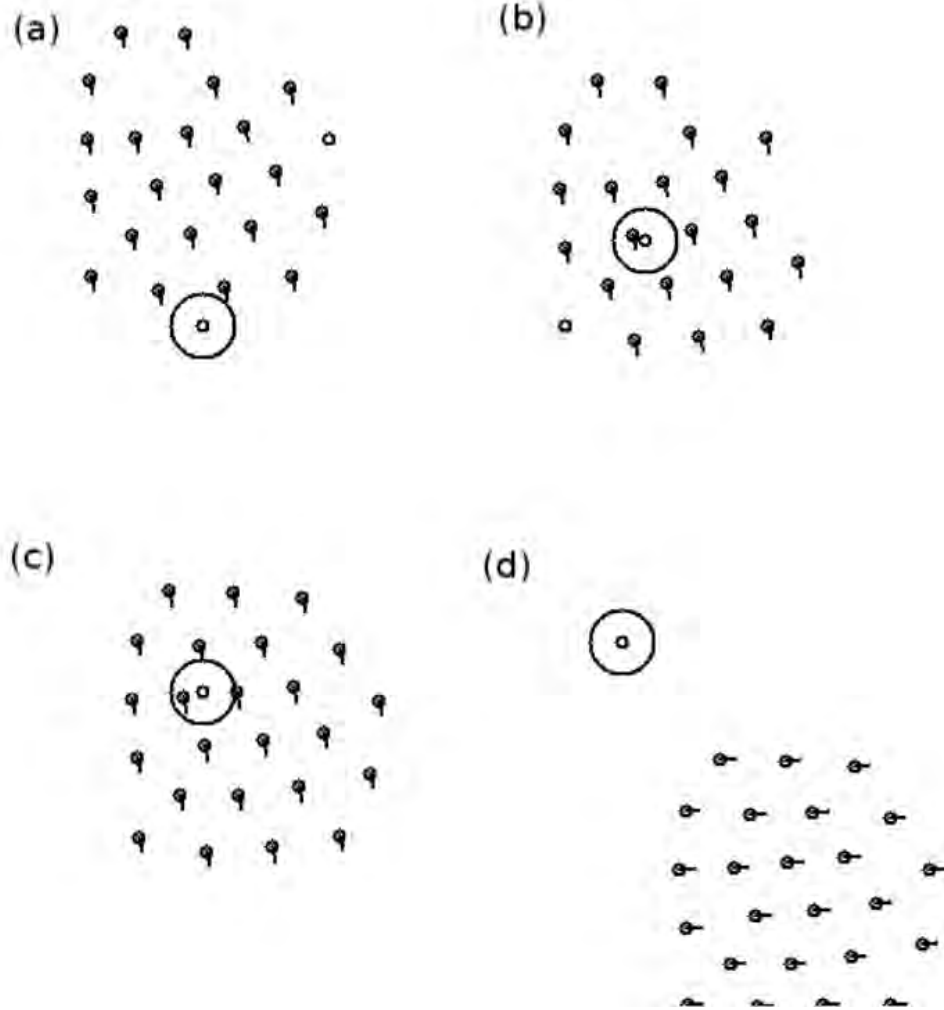
Figure 3.6: When r_s is too small compared to D, the swarm does not detect the target. The snapshots (a)-(d) show the swarm flying over the target without locating it. Here $N = 24$, $r_s = 1.5$, required percentage for consensus is $p = 25\%$, and D stabilizes at ≈ 13.5.

3.2.3   An Approximation for the Optimal Swarm Diameter

Now that we have an upper bound on D_{opt}, we assume that condition (3.2) is met and look for approximate expressions for D_{opt} and $\bar{T}$. We note that the area of overlap A_o between the target and swarm areas, when the centers are separated by a distance d, is given by

$$A_o = r_s^2 \arccos \left[\frac{z}{r_s} \right] + \frac{D^2}{4} \arccos \left[\frac{2(d-z)}{D} \right] - $$
$$z\sqrt{r_s^2 - z^2} - (d-z)\sqrt{D^2/4 - (d-z)^2} \ , \quad (3.3)$$

where

$$z \equiv \frac{r_s^2 - D^2/4 + d^2}{2d} \ . \tag{3.4}$$

Eq. 3.3 is valid for $|r_s - D/2| \le d \le r_s + D/2$. Now, as above, we require that A_o be at least equal to 25% of the swarm area in order for the target to be detected. Thus, we obtain an implicit equation for the maximum separation d_{max} between the center of the swarm and the target location such that the target is detected:

$$A_o(r_s, D, d_{\mathrm{max}}) = \pi D^2/16 \ . \tag{3.5}$$

The parameter d_{max} will depend therefore upon r_s and D. At least in terms of the time spent within the searching phase of the algorithm, the shortest time $\bar{T}$ until detection ought to occur when, for a given r_s, D is chosen such that d_{max} is maximized (see Fig. 3.7), giving the largest effective target size to hit; hence this D should be D_{opt}. Furthermore, we expect a scaling law such that the time to detection is roughly given by

$$\bar{T} \approx \tau \left[\frac{A_{\mathrm{field}}}{\pi d_{\mathrm{max}}^2} - 1 \right] \ , \tag{3.6}$$

where τ is a characteristic timescale and A_{field} is the total area of the search field.
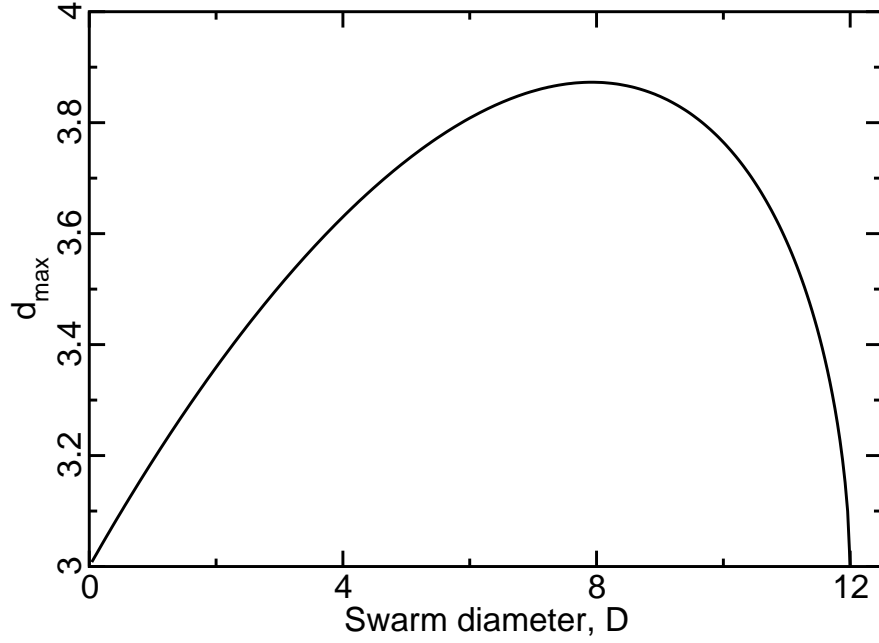
Figure 3.7: Parameter $d_{\max}$ versus D for $r_s = 3.0$. Note that $d_{\max}$ increases at first, as the growing size of the swarm allows it to be further away while still easily satisfying (3.2). However, after the peak at $D_{\mathrm{opt}} \approx 8$, the condition (3.2) becomes the limiting factor, requiring greater overlap between the two areas for detection to occur.

We have experimentally verified this scaling, with numerical results usually quite close to the theoretical values, as illustrated by the example with $r_s = 3.0$ in Fig. 3.4. We note that the actual time to detection is a bit above the theory for $D > D_{\text{opt}}$, presumably due to our assumption of constant density in deriving Eq. 3.5; that is, (especially for large D) the area of overlap between target and swarm may be sufficient, but still not contain at least 25% of the agents, causing the time to detection to be above that expected.

3.3   Concluding Remarks

We considered a mine counter-measure scenario using multiple agents that move cooperatively via swarming. The agents use a variety of signal filters to determine when they are within sensing range of a target and to reduce noise for more accurate control and position estimation of targets. We explored the parameter space through simulations, determining optimal values for some of the search parameters. We derived scaling properties of the system, compared with the data from simulations, and found a good experimental-analytical fit.

There are many openings for future research. First, we could use alternate methods in some parts of the algorithm. A potential change is to use a compressed sensing method [COS09] instead of least-squares for estimating a target's location, which would enable us to find multiple overlapping targets at the same time. Another interesting modification would be to use an anisotropic Lévy search, and take previously covered paths into account. Different scenarios could also be evaluated, which might lead to different results for accuracy and efficiency, or even suggest using new algorithms. For example, we could extend the two dimensional problem to 3-D, as would be the case for underwater targets. Or, perhaps the model for the detected signal is unknown, in which case we would

employ a different method to estimate the target positions. Finally, apart from numerical simulations, we plan to do experiments on a real testbed, with small robotic vehicles as agents. This would provide an evaluation of the algorithm in the presence of real sensor noise, which may not be entirely Gaussian in nature.

Diffuse interface surface tension models in an expanding flow

CHAPTER 4

Introduction to the Surface Tension Models

4.1 Background

There is a need to develop simple computational models for surface tension in the droplet breakup phenomena. As an example, consider a piece of material that expands under a sudden pulse of energy that comes from laser fusion [Ka10] or heavy ion fusion [Fa08]. The material will breakup, and surface tension plays an important role in the ensuing dynamics. There are many numerical methods that deal with surface tension in two-phase fluids. This problem is known for its computational stiffness. It contains two different time scales, the small surface tension time scale and the convection time scale. Three main algorithms exist for two-phase fluids. The sharp interface method tracks the interface explicitly, yet it requires extensive processing when the interface splits and merges. Since droplet breakup involves mainly merging and splitting of the interface, we do not consider sharp interface methods in this chapter. The level-set algorithm uses a implicit surface function to track the boundary. The diffusive interface algorithm uses a phase variable to describe the transition between materials. These algorithms have been studied theoretically and numerically with many variants. For some previous results, see Fig. 4.1 through 4.3

The basic level-set model for two immiscible fluids uses a function ϕ, where $\phi = 0$ denotes the boundary between the two fluids. Among the first to propose

this model is [SSO94], which combines the Navier-Stokes equation for two fluids with a force at the interface.

$$\rho\frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \nabla)\vec{V} = -\nabla p + \nabla \cdot (2\nu D) - \tau \kappa(\phi)\nabla H(\phi) + f. \qquad (4.1)$$

This equation is then coupled with the level set equation for the interface:

$$\phi_t + \nabla \cdot (\phi \vec{V}) = 0. \qquad (4.2)$$

In this model, $\vec{V}$ is the velocity field, D is the deformation tensor $\frac{1}{2}(\nabla\vec{V}+\nabla\vec{V}^T) - \frac{1}{3}\nabla\vec{V}I$, p is the pressure and f denotes external force. These parameters are the same as the original Navier-Stokes model. $\kappa(\phi) = \nabla \cdot \frac{\nabla\phi}{|\nabla\phi|}$ is the curvature of the boundary, τ is the surface tension coefficient, and H is the Heaviside function, or in the numerical implementation, a smoothed Heaviside function. (4.1) is the Navier-Stokes equation with a surface tension term $\kappa\vec{n}\delta(d)$, where $\vec{n}$ is unit outward normal vector at the front, d is normal distance to the front, and δ is the Dirac delta function. Recent models are designed to improve computational speed [SSH07, SO09]. The level-set model can be naturally modified for a compressible flow, with the price of a more complicated set of equations, e.g. [CFA01].

The original Cahn-Hilliard equation [CH58], together with the Allen-Cahn equation are one of the most well-known dynamic models for diffuse interface dynamics associated with surface energies. The Cahn-Hilliard equation can be written as an H^{-1} gradient descent for a Ginzburg-Landau free energy $E(u)$:

$$u_t = \Delta(\frac{\delta E(u)}{\delta u}) + \lambda u, \qquad (4.3)$$

where

$$E(u) = \int (\epsilon|\nabla u|^2 + \frac{1}{\epsilon}g(u)), \qquad (4.4)$$

and $g(u)$ is a double-well potential that characterizes the two phases. It is normally taken as an even-order polynomial, for example

$$g(u) = u^2(1-u)^2, \qquad (4.5)$$

in which case $f(u) = g'(u) = 2u(1-u)(1-2u)$. The Allen-Cahn equation, on the other hand, is L^2 gradient descent for the same energy. Papers such as [Peg89, ABC94, CS06] analyze the convergence and stability of the Cahn-Hilliard equation.

The combination of Cahn-Hilliard dynamics and fluid mechanics give rise to several related models for fluid interfaces. For example, the incompressible Navier-Stokes-Cahn-Hilliard model [BJO96, Boy99, Jac99, KKL04, YFL05] couples the incompressible fluid mechanics with a diffuse interface model.

$$\rho(\frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \nabla)\vec{V}) = -\nabla p + \nabla \cdot (2\nu D) - \nabla \cdot (\epsilon \rho \nabla u \otimes \nabla u) + f, \tag{4.6}$$

$$\nabla \cdot \vec{V} = 0, \tag{4.7}$$

$$\rho(\frac{\partial u}{\partial t} + \vec{V} \cdot \nabla u) = \Delta K(u), \tag{4.8}$$

$$K(u) = \frac{\partial g(u)}{\partial u} - \frac{1}{\rho}\Delta u. \tag{4.9}$$

The $\nabla \cdot (\epsilon \rho \nabla u \otimes \nabla u)$ term in (4.6) represents surface tension. The additional advection term represents the mechanics of fluid flow.

One can modify the above model to include compressible fluids by replacing equation (4.7) with

$$\frac{\partial \rho}{\partial t} + \vec{V} \cdot \nabla \rho = 0, \tag{4.10}$$

see [LT98, AF08, FPR10]. Other models include [PPD94], which proposes the Cahn-Hilliard type model under a gravitational field. The Allen-Cahn model with a transport term has been studied in [LST10] in the context of the sharp interface limit yielding motion by mean curvature plus a transport term.

It should be noticed that the level-set model and the Cahn-Hilliard model have mathematical relationships. When $\epsilon \to 0$, the Ginzburg-Landau energy (4.4) Γ-converges to the surface energy $\int |\nabla u|$, which can be considered as the surface tension related energy in the level set model [KS89, CS06, LGB07].

In this chapter, we focus on the movement of a small droplet of incompressible material within another compressible fluid. It is an important model problem for a full-scale numerical simulations for material breakup. We take a simpler model to that of the compressible Navier-Stokes-Cahn-Hilliard model [AF08]. Instead of having the velocity field satisfying a Navier-Stokes equation, we consider the same model under a specified velocity field which may not be divergent free, and focus on the droplet breakup phenomena.

In the next section we present the specific models, namely the Cahn-Hilliard equation with advection and Allen-Cahn equation with advection. Section 3 analyzes the basic property and droplet breakup condition of the Cahn-Hilliard equation with advection. Section 4 analyzes the Allen-Cahn equation with advection. Section 5 shows numerical simulation results for both models.

4.2   The Model Problem

We consider the following model for a diffusive interface with advection [LT98],

$$u_t + \nabla \cdot (u\vec{V}) = \mathcal{F}(u), \tag{4.11}$$

where $\vec{V}$ is the prescribed external flow field and $\mathcal{F}(u)$ represents the surface tension force. When $\vec{V} = 0$, we obtain the original diffuse interface equation. Our main interest is when $\vec{V}$ is expanding, or problems in which $\nabla \cdot \vec{V} \neq 0$ in general. We note that the incompressible case is well studied, however the compressible case less so. For simplicity we choose the Neumann condition $\frac{\partial u}{\partial n} = 0$ on $\partial\Omega$. All the models considered in this chapter are of the form (4.11) with the $\mathcal{F}$ term related to Ginzburg-Landau energy (4.4).

4.2.1 The advective Cahn-Hilliard equation

The original Cahn-Hilliard equation comes from a phase separation problem. It is a non-local Mullins-Sekerka flow for $E(u)$ [Gra89, Peg89, ABC94] .

$$\mathcal{F}(u) = \Delta(\frac{\delta E(u)}{\delta u}).$$
(4.12)

Thus, the equation can be written as

$$u_t + \nabla \cdot (u\vec{V}) = \Delta K(u),$$
(4.13)

where

$$K(u) = -\epsilon \Delta u + \frac{1}{\epsilon} f(u).$$
(4.14)

4.2.2 The advective Allen-Cahn equation

In the Allen-Cahn equation, the surface tension term is a mean curvature flow for the energy $E(u)$. Thus, the equation can be written as

$$u_t + \nabla \cdot (u\vec{V}) = -K(u),$$
(4.15)

with the same K as in (4.14).

4.2.3 The advective Allen-Cahn equation with mass conservation

If we integrate the original Allen-Cahn equation, we can see that it does not automatically conserve mass. Thus, an additional term λ is often added to the equation for this reason [RS92]. We can add a similar term here, but we would like to add λu instead of λ to keep u localized. The equation can be written as

$$u_t + \nabla \cdot (u\vec{V}) = -K(u) + \lambda u,$$
(4.16)

where λ is chosen so that $\int_\Omega u$ is a constant M. Or, as we can compute,

$$\lambda = \frac{\int_\Omega K(u)}{M} = -\frac{1}{\epsilon} \frac{\int_\Omega f(u)}{M}.$$
(4.17)

These three equations are the main focus of this chapter. From now on we call them the advective Cahn-Hilliard equation, the advective Allen-Cahn equation and the advective nonlocal Allen-Cahn equation, respectively.

Properties of $\vec{V}$ play an important role here. In papers like [Boy99, KKL04], the velocity field satisfies a Navier-Stokes equation, thus the velocity field $\vec{V}$ is divergent free. In the situation of our main concern, $\vec{V}$ is not divergent free. The flow is expanding where $\nabla \cdot \vec{V} > 0$ and contracting where $\nabla \cdot \vec{V} < 0$.

Unlike their nonadvective counterpart, mass conservation is not automatically satisfied in these advective equations. For example, if we integrate (4.13), we would have

$$\left(\int_\Omega u \right)_t + \int_{\partial\Omega} u\vec{V} \cdot \vec{n} = \int_{\partial\Omega} \nabla K(u) \cdot \vec{n}. \tag{4.18}$$

Under Neumann boundary condition, the right hand side become 0. Only when we exert a no-flow condition $\vec{V} \cdot \vec{n} = 0$ on the boundary can we have mass conservation. In fact, this no-flow condition would simplify many proofs below. We assume this is satisfied by having a small layer of $\vec{V}$ that vanishes near the boundary. We also assume $\vec{V}$ is smooth enough in the following arguments.
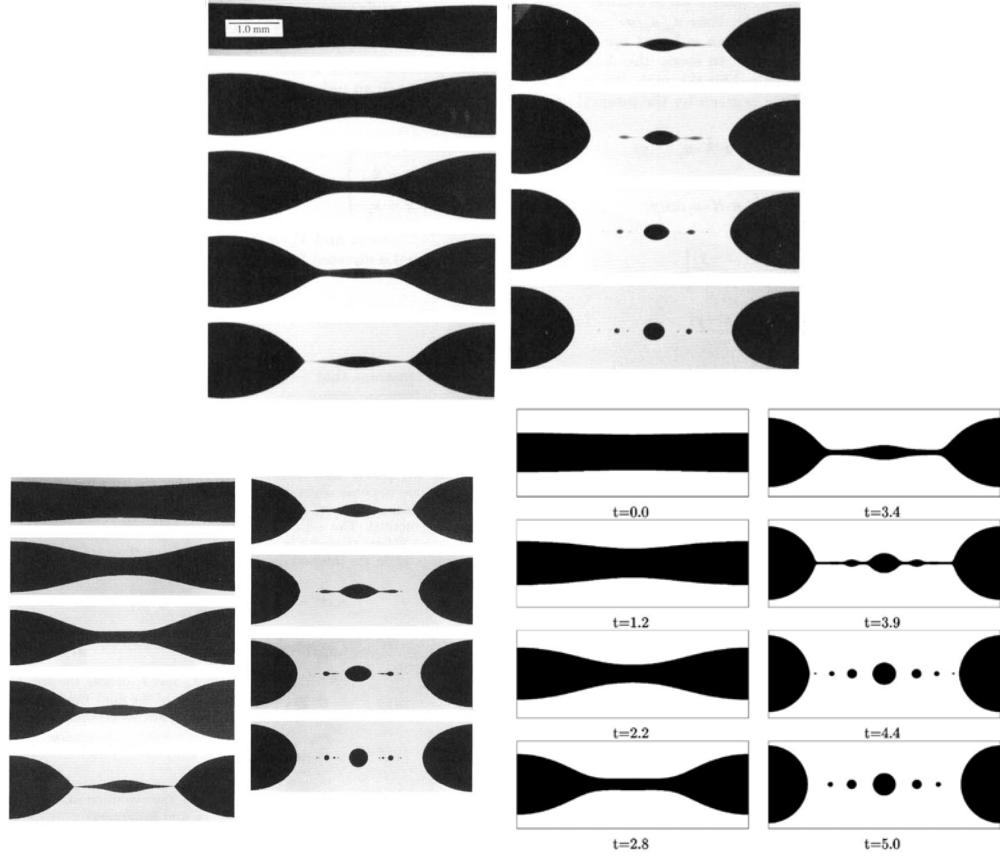
Figure 4.1: Up: Experiment result of a liquid/liquid thread undergoing Rayleigh instability. Left: Numerical simulation using sharp interface method, both from [TSO92] copyright 1992 Cambridge University Press, reproduced with permission. Right: Numerical simulation using diffuse interface method, from [KKL04], copyright 2004 Elsevier, reproduced with permission.
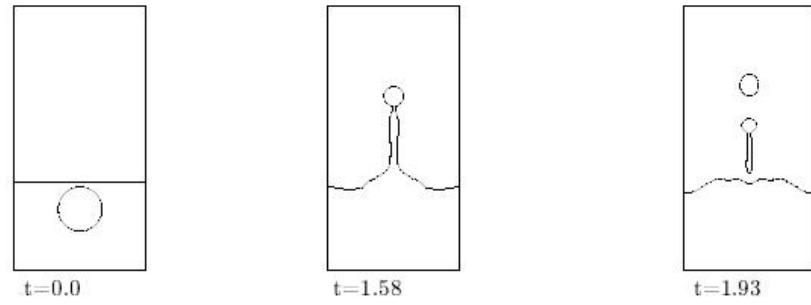
Figure 4.2: A level-set simulation of air bubble bursting at surface, from [SSO94], copyright 1994 Elsevier, reproduced with permission.
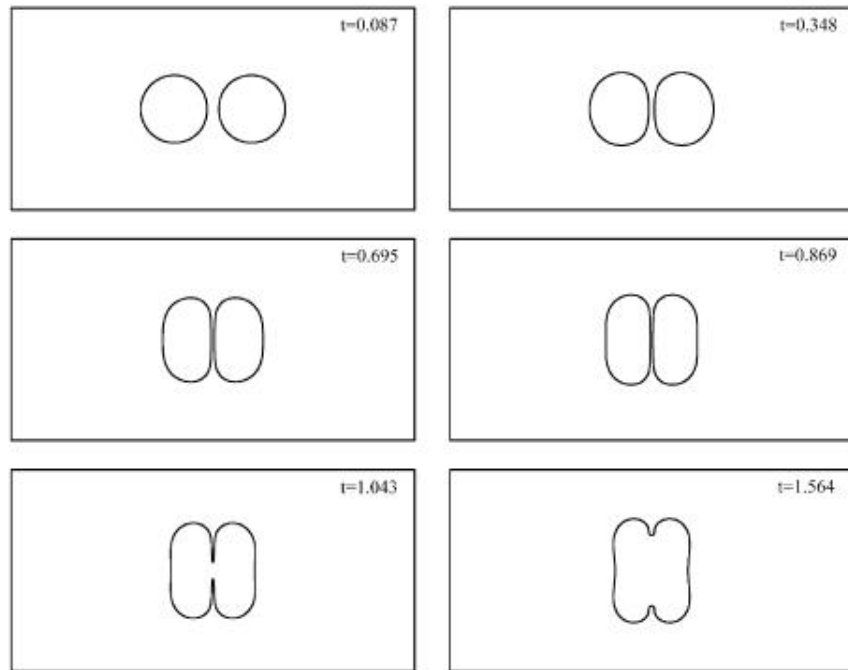
Figure 4.3: A diffuse interface simulation on two drops collide, from [YFL05], copyright 2005 Elsevier, reproduced with permission.

CHAPTER 5

Analytical Properties of the Model Problem

5.1 Property of the advective Cahn-Hilliard equation

In this section we prove basic properties of the advective Cahn-Hilliard equation. We begin with the existence and uniqueness property of the equation, then move on to the analysis of the breakup condition.

5.1.1 Existence and uniqueness

The following existence and uniqueness theorem is similar to that of the original Cahn-Hilliard equation [Tem88]. In the proof below and related arguments, the symbol C denotes a generic constant.

Theorem 5.1.1 *If $g(u)$ in (4.4) is a polynomial of order 2p, for every given u_0 in $L^2(\Omega)$, the equation (4.13) with $u(0) = u_0$ has a unique solution u that belongs to $C([0,T]; L^2(\Omega)) \cap L^2(0,T; H_0^2\Omega) \cap L^{2p}(0,T; L^{2p}(\Omega)), \forall T > 0$.*

The proof for this theorem follows the same step as the Galerkin method for proving other equations like the Navier-Stokes equation and original Cahn-Hilliard equation, with the only difference in the a priori estimate. See [Tem95, Tem88]. We only present the different a priori estimates here.

The weak form of (4.13) is

$$(u'(t), w) + \epsilon A(u, w) + B(\vec{V}, u, w) + \frac{1}{\epsilon}(f'(u)\nabla u, \nabla w) = 0, \forall w \in H^2(\Omega), \quad (5.1)$$

where $A(u, w) = (\Delta u, \Delta w)$, $B(\vec{V}, u, w) = \int_\Omega \nabla \cdot (u\vec{V})w$.

Taking $w = u$ we have

$$\frac{1}{2}\frac{d}{dt}|u|^2 + \epsilon|\Delta u|^2 + \frac{1}{\epsilon}(f'(u)\nabla u, \nabla u) + \frac{1}{2}(|u|^2, \nabla \cdot \vec{V}) = 0. \quad (5.2)$$

Since $f'(s) \geq b_{2p}s^{2p-2} - C$,

$$\frac{1}{2}\frac{d}{dt}|u|^2 + \epsilon|\Delta u|^2 + \frac{1}{\epsilon}\int_\Omega (b_{2p}u^{2p-2}|\nabla u|^2) \leq C|\nabla u|^2 + \frac{C}{2}|u|^2. \quad (5.3)$$

Thus we can get the upper bound for u in $L^2(0, T; H^2(\Omega))$. To get the upper bound of u in $L^\infty(0, T; L^2(\Omega))$, we see that

$$\begin{aligned} C|\nabla u|^2 &\leq C|u|\, \|u\|_{H^2(\Omega)} \\ &\leq C|u|(|\Delta u| + M) \\ &\leq \frac{\epsilon}{2}|\Delta u|^2 + C|u|^2 + CM^2, \end{aligned} \quad (5.4)$$

where $M = \int_\Omega u$ is the total mass. Thus,

$$\frac{d}{dt}|u|^2 + \epsilon|\Delta u|^2 + \int_\Omega (b_{2p}u^{2p-2}|\nabla u|^2) \leq C|u|^2 + CM^2. \quad (5.5)$$

The rest of the proof are similar to [Tem95, Tem88]. By using the Gronwall inequality we get an upper bound for u in $L^\infty(0, T; L^2(\Omega))$. This suffices to show the continuity and uniqueness.

5.1.2 Energy estimate

The original Cahn-Hilliard equation has an energy term that serves as a Lyapunov function:

$$J(u) = \int_\Omega \frac{\epsilon}{2}|\nabla u|^2 + \frac{1}{\epsilon}g(u). \quad (5.6)$$

This term can be estimated by multiplying $K(u)$ on both sides of (4.13) then integrate by parts. Following the same pattern, we get

$$J(u)_t + (\nabla \cdot (u\vec{V}), K(u)) = -|\nabla K(u)|^2. \tag{5.7}$$

We can estimate the new term by

$$(\nabla \cdot (u\vec{V}), K(u)) = (\vec{V} \cdot \nabla u, K(u)) + ((\nabla \cdot \vec{V})u, K(u)). \tag{5.8}$$

The first term

$$(\vec{V} \cdot \nabla u, K(u)) = -(\nabla \cdot \vec{V}, \frac{\epsilon}{2}|\nabla u|^2 + \frac{1}{\epsilon}g(u)) - \epsilon \int_\Omega (\nabla u)^T \nabla \vec{V} \nabla u, \tag{5.9}$$

due to the fact that

$$\int_\Omega \vec{V} \cdot \nabla u \Delta u = \int_\Omega \vec{V} \cdot (\nabla \cdot (\nabla u \otimes \nabla u) - \frac{1}{2}\nabla(|\nabla u|^2))$$

$$= \int_\Omega -\nabla \vec{V} : (\nabla u \otimes \nabla u) + \frac{1}{2}\nabla \cdot \vec{V}|\nabla u|^2 \tag{5.10}$$

$$= \int_\Omega -(\nabla u)^T \nabla \vec{V} \nabla u + \frac{1}{2}\nabla \cdot \vec{V}|\nabla u|^2.$$

The right-hand side of (5.9) is bounded from below by $-2||\nabla \vec{V}||_{L^\infty} J(u)$.

The second term of (5.8) is bounded by

$$((\nabla \cdot \vec{V})u, K(u)) \geq -\epsilon(\frac{1}{2}\int u^2 \Delta \nabla \cdot \vec{V} - \int |\nabla u|^2 \nabla \cdot \vec{V}) - \frac{||\nabla \cdot \vec{V}||_{L^\infty}}{\epsilon} \int |uf(u)|$$

$$\geq -C(J(u) + |u|_2^2). \tag{5.11}$$

Putting everything together, we have

$$J(u)_t \leq C(J(U) + |u|_2^2), \tag{5.12}$$

which, using Gronwall's inequality and the bound of $|u|_2^2$ above gives

$$J(u)_t \leq \exp(Ct)J(u_0) + C. \tag{5.13}$$

We can see that, the energy of u is bounded at every finite time interval $[0, T]$ and increases at most exponentially.

5.1.3 Droplet breakup

When the the external flow field is sufficiently large, the advective Cahn-Hilliard model exhibits droplet breakup as illustrated in Fig. 5.1. Similar phenomenon have been observed in numerous reaction-diffusion systems, see for example [NU99], [KWW07], [KO94] and references therein. In this section we perform a detailed study of the breakup phenomenon for the advective Cahn-Hilliard model in one dimension. We recall that the one dimensional case of (4.13) is:

$$u_t + (V(x)u)_x = K_{xx}; \quad K = -\varepsilon u'' + \frac{1}{\varepsilon}f(u). \tag{5.14}$$

We choose a specific form of $f(u)$ in our discussion:

$$f(u) = 2u(1-u)(1-2u). \tag{5.15}$$

Other forms of $f(u)$ follow a similar discussion. In [NU99], Nishiura and Ueyema proposed a set of conditions for the occurrence of self-replication in reaction-diffusion models. Roughly stated, they are are:

1. The disappearance of the steady state due to a fold-point (or saddle-node) bifurcation.

2. The existence of the so-called dimple-eigenfunction at the threshold, which is responsible for the initiation of the breakup process.

3. The steady state is stable on one side of the fold point and is unstable on the other.

The importance of these conditions is that the breakup of a droplet can be understood in terms of the analysis of the steady state solution of (5.14) which satisfies

$$(V(x)u)_x = K_{xx}; \quad K = -\varepsilon u'' + \frac{1}{\varepsilon}f(u). \tag{5.16}$$

The breakup analysis for (5.14) is very similar to [KWW07] where the Brusselator and other reaction-diffusion systems having mesa-type structures were shown to exhibit self-replication. For simplicity, we will only consider a special case

$$V(x) = \frac{V_0}{\varepsilon}x, \tag{5.17}$$

and get the following asymptotic result:

Result 5.1.2 *Consider (5.14) in the limit $\varepsilon \ll 1$, with $V(x)$ given by (5.17), and with even initial conditions for u. For a given mass $M = \int_{-\infty}^{\infty} u\,dx$, let*

$$V_c = \frac{V_{c0}}{M^2} \tag{5.18}$$

where $V_{c0} \approx 1.326$ is a constant whose precise value is given below in (5.31). If $V < V_c$ then there exists a steady state $u(x,t) = u(x)$ in the form of a droplet. If $V > V_c$, no such steady state exists. As V is slowly increased past V_c, the dropet will split in the middle and breakup into two droplets.

The derivation of this result consists of an analytic verification of the Nishiura-Ueyema conditions 1 and 2. Due to space limitations, we omit the verification of Condition 3 but refer the reader to [KWW07] where this condition 3 is proved for a similar model.

Verification of Nishiura-Ueyema condition 1. We seek a steady state solution $u(x)$ which is even. It then follows that K is also even and upon integrating (5.16) on the interval $[0, x]$, we obtain

$$K_x = \frac{V_0}{\varepsilon}xu.$$

We now change variables $K = \frac{1}{\varepsilon}w$ to obtain a system

$$w_x = V_0 xu; \qquad -\varepsilon^2 u_{xx} + f(u) = w. \tag{5.19}$$
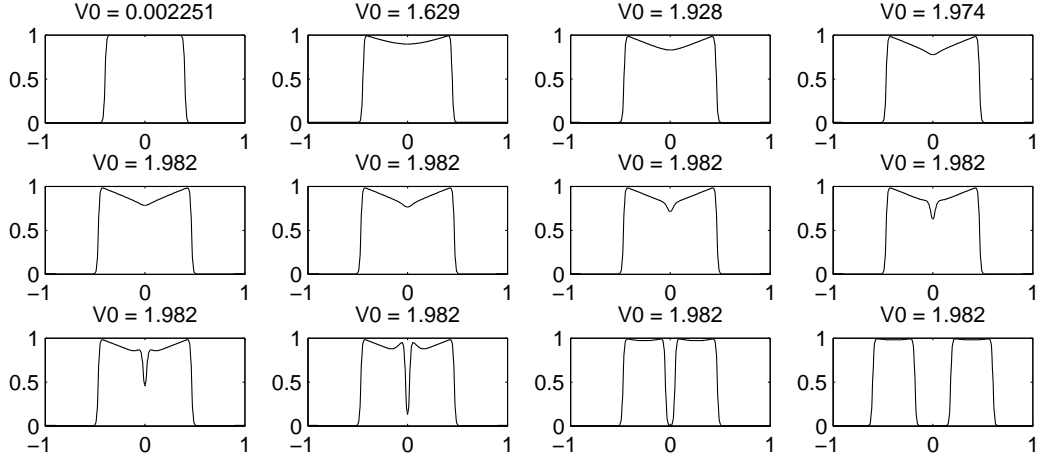
Figure 5.1: Snapshots of temporal dynamics of (5.16) with V given by (5.17). Here, $\varepsilon = 0.01$ and the mass of the droplet is taken to be $M = \int_{-\infty}^{\infty} u\,dx = 0.8$. The parameter V_0 is slowly increased in time according to the formula $V_0 = 0.001t$.

Since we assumed that u is even, we consider only the half-line $x \geq 0$; the boundary conditions become

$$u'(0) = 0 = u'(\infty); \quad \int_0^{\infty} u = M/2 \tag{5.20}$$

where M is a given total mass of u. Since the time-dependent PDE (5.14) conserves the mass of u, M is also the initial mass of $u(x,t)$ at $t = 0$.

We will construct a solution to (5.19, 5.20) for which $u(x)$ has a sharp interface located at some position $x = l > 0$ with $u \sim 0$ for $x > l$. Some typical such profiles for $u(x)$ are shown in Fig. 5.2. Such a solution has a transition layer consisting of the interface near $x = l$ and an outer region to the left of $x = l$. In the transition layer, we rescale the space variable

$$x = l + \varepsilon y; \quad u(x) = U(y); \quad w(x) = W(y)$$

to obtain

$$W_y \sim \varepsilon^2 V_0 y U.$$

51
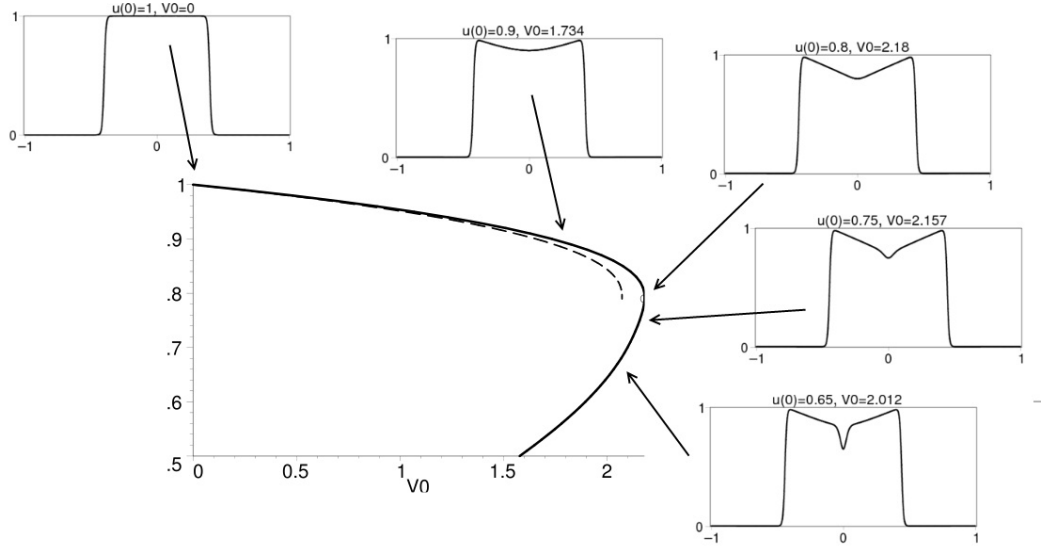
Figure 5.2: The bifurcation structure of the the steady state equations (5.19,5.20) with $\varepsilon = 0.01$, $M = 0.8$; V_0 is plotted vs. $u(0)$. Solid curve represents the solution to the full system numerically; the dotted curve is the asymptotic formula (5.29). The coordinates of the fold point are $u(0) = 0.79$, $V_0 = 2.18$. The inserts show the profile of $u(x)$ for selected points along the bifurcation curve as indicated.

To leading order, we have $W_y \sim 0$ so that $W \sim W_0$ is constant. We then obtain an ODE for U,

$$U_{yy} + f(U) - W_0 = 0. \tag{5.21}$$

The interface corresponds to a heteroclinic orbit of the ODE (5.21) which connects the two saddle equilibria of (5.21). Such heteroclinic connection exists if and only if $\int_{U_-}^{U_+} [f(U) - W_0] \, dU = 0$, where $U_\pm$ are the equilibria points that satisfy $f(U_\pm) - W_0 = 0$ with $U_+ \neq U_-$. Using $f(u) = 2u(1-u)(1-2u)$, this yields $U_+ = 1$, $U_- = 0$ and $W_0 = 0$; the explicit solution for $U(y)$ is then given by

$$U(y) = \frac{1}{2}\left(1 - \tanh(y/\sqrt{2})\right)$$

52

with

$$U(+\infty) = 0; \quad U(-\infty) = 1.$$

In the outer region away from the interface, to leading order we have

$$f(u) \sim w, \quad 0 \le x < l. \tag{5.22}$$

Substituting (5.22) into (5.19) we then obtain

$$\frac{du}{dx} = xV_0 \frac{u}{f'(u)}; \quad u(l) = 1. \tag{5.23}$$

The boundary condition is obtained from matching to the outer solution, $u(l) = U(-\infty) = 1$. The solution to (5.23) is given by

$$\frac{V_0}{2} x^2 = \int_{u_0}^{u} \frac{f'(s)}{s} ds; \quad x < l \tag{5.24}$$

where $u_0 = u(0)$. Thus we obtain the following relationship between l and u_0,

$$\frac{V_0}{2} l^2 = G(u_0) \tag{5.25}$$

where

$$G(u_0) := \int_{u_0}^{1} \frac{f'(s)}{s} ds = -6u_0^2 + 12u_0 - 2\ln u_0 - 6.$$

It remains to relate l to M. Since $u \sim 0$ to the right of the interface, the mass of u is asymptotically given by

$$\frac{M}{2} \sim \int_0^l u(x) dx \tag{5.26}$$

where we ignored the $O(\varepsilon)$ contribution to the mass from the interface. Writing (5.24) as

$$x^2 = \frac{2}{V_0} (G(u_0) - G(u))$$

53

and substituting into (5.26) we obtain

$$\frac{M}{2} \sim \left(\frac{2}{V_0}\right)^{1/2} \int_{u_0}^{1} u \frac{d}{du} \sqrt{G(u_0) - G(u)} du \tag{5.27}$$

$$\sim \left(\frac{2}{V_0}\right)^{1/2} \left\{ \sqrt{G(u_0)} - \int_{u_0}^{1} \sqrt{G(u_0) - G(u)} du \right\}. \tag{5.28}$$

so that

$$V_0 \sim \frac{1}{M^2} 8 \left(\sqrt{G(u_0)} - \int_{u_0}^{1} \sqrt{G(u_0) - G(u)} du \right)^2. \tag{5.29}$$

Next, note that $G(1) = 0$ and $G'(u_0) = -f'(u_0)/u_0$; in particular $G(u_0)$ attains a maximum at u_m which satisfies $f'(u_m) = 0$:

$$u_m := \frac{3 + \sqrt{3}}{6} = 0.78868. \tag{5.30}$$

It follows that the solution to the outer problem (5.23) only exists if $u_m < u(0) < 1$. In terms of M, the critical threshold for existence is obtained by substituting $u_0 = u_m$ into (5.29); namely $V_c = \frac{V_{c0}}{M^2}$ where the constant V_{c0} is given by

$$V_{c0} := 8 \left(\sqrt{G(u_m)} - \int_{u_m}^{1} \sqrt{G(u_m) - G(u)} du \right)^2 \approx 1.32606. \tag{5.31}$$

This shows the existence of the fold-point for V_0 as given by Result 5.1.2.

Verification of Nishiura-Ueyema condition 2. Here, we follow closely an analogous derivation in [KWW07]. The key is to demonstrate that when V_0 is close to the threshold value V_c, an additional boundary layer in the shape of an inverted spike forms at the center of the droplet. To see this, suppose that V_0 is sufficiently close to V_c so that near $x = 0$, we may expand

$$u(x) \sim u_m + \delta u_1(x), \quad w \sim f(u_m) + \delta^2 w_1 + ...; \quad \delta \ll 1. \tag{5.32}$$

The small parameter δ will be related to ε below. The equation for w_1 then becomes

$$\delta^2 w_{1x} \sim V_0 x u_m$$
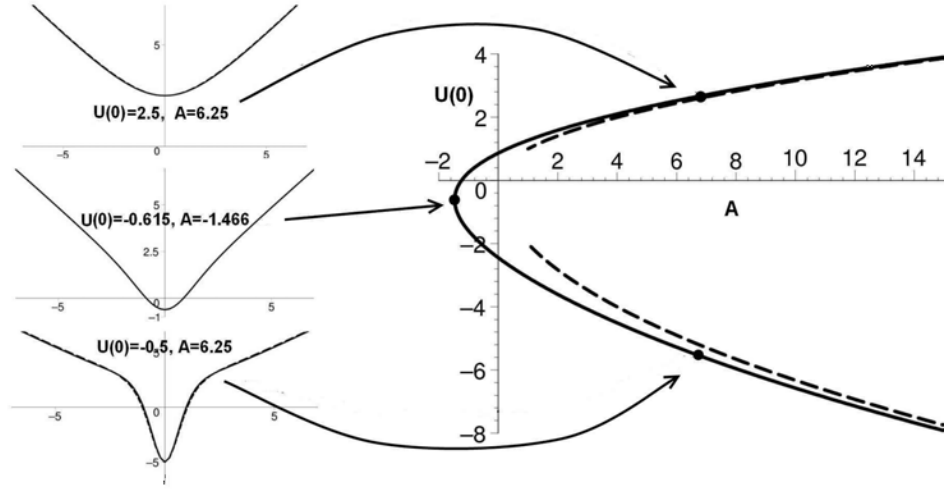
Figure 5.3: Right side: Bifurcation diagram A vs. $U(0)$ for the core problem (5.41). Solid curve is the numerical solution to (5.41); dashed lines represent the asymptotics for large A as given by (5.43,5.42). Left: the solution profiles with $U(0)$, A as indicated.

so that

$$w_1(x) \sim w_1(0) + \delta^{-2}\frac{V_0 u_m}{2}x^2. \tag{5.33}$$

The consistency condition for (5.33) is that $x/\delta \ll \infty$; this will be satisfied below. We now expand in Taylor series

$$f(u) - w \sim -w_1(0)\delta^2 - \frac{V_0 u_m}{2}x^2 + u_1^2\frac{f''(u_m)}{2}\delta^2 \tag{5.34}$$

where we recall that $f'(u_m) = 0$. Substituting (5.33, 5.34) into (5.19) we obtain

$$\varepsilon^2 u_{1xx} - u_1^2\frac{f''(u_m)}{2}\delta^2 + w_1(0)\delta^2 + \frac{V_0 u_m}{2}x^2 = 0. \tag{5.35}$$

To determine the right scaling for δ, rescale

$$x = \alpha z, \ u_1(x) = U(z)$$

so that (5.35) becomes

$$U_{zz} - \left(\frac{f''(u_m)}{2}\frac{\delta^2\alpha^2}{\varepsilon^2}\right)U^2 + \left(\frac{V_0 u_m}{2}\frac{\alpha^4}{\varepsilon^2}\right)z^2 + w_1(0)\frac{\delta^2}{\varepsilon^2}\alpha^2 = 0 \qquad (5.36)$$

We now choose α, δ so that (5.36) becomes

$$U_{zz} = U^2 - z^2 - A \qquad (5.37)$$

i.e.

$$\alpha := \varepsilon^{1/2}\left(\frac{V_0 u_m}{2}\right)^{-1/4}; \quad \delta := \varepsilon^{1/2}\left(\frac{V_0 u_m}{2}\right)^{1/4}\left(\frac{f''(u_m)}{2}\right)^{-1/2} \qquad (5.38)$$

$$A := \frac{2w_1(0)}{f''(u_m)}. \qquad (5.39)$$

Matching with the outer solution, in the limit $z \to \infty$ we impose the boundary condition $u_{xx}\varepsilon^2 \ll 1$; or $U_{zz} \sim 0$. Thus the boundary conditions for (5.37) becomes

$$U_z(0) = 0; \quad U \sim z \text{ as } z \to \infty. \qquad (5.40)$$

The equations (5.37) and (5.40) together comprise the *core problem* which fully describes the growth of the inverted spike at the origin. The scaling $\alpha = O(\varepsilon^{1/2})$ quantifies the width of the the core spike in terms of the $O(\varepsilon)$ interface width. This core problem is identical to the core problem for the Brusselator and other reaction-diffusion systems; we refer the reader to [KWW07] for details. For convenience, we state the main result about (5.37, 5.40) as derived in [KWW07]:

Lemma 5.1.3 (from [KWW07], Theorem 2) *Consider the core problem*

$$U_{zz} = U^2 - z^2 - A; \quad U_z(0) = 0; \quad U \sim z \text{ as } z \to \infty. \qquad (5.41)$$

There exists a constant A_c such that (5.41) has precisely two monotone solutions for $A > A_c$ and no monotone solutions when $A < A_c$.
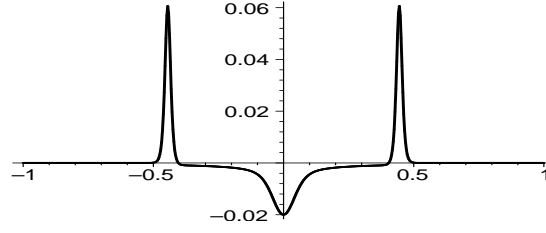
Figure 5.4: The shape of the the eigenfunction corresponding to the zero eigenvalue at the fold point of the bifurcation diagram for (5.19,5.20) with $\varepsilon = 0.01$, $M = 0.8$. The parameter $V_0 = 2.18$ is chosen to be at the fold point.

When $A \gg 1$, equation (5.41) admits two monotone solutions $U^{\pm}(z)$ with the following uniform asymptotic expansions:

$$U^+(z) \sim \sqrt{A + z^2}, \quad with \quad U^+(0) \sim \sqrt{A}; \tag{5.42}$$

$$U^-(z) \sim \sqrt{A + z^2}\left(1 - 3\operatorname{sech}^2\left(\frac{A^{1/2}z}{\sqrt{2}}\right)\right), \quad with \quad U^+(0) \sim -2\sqrt{A}; \tag{5.43}$$

For any monotone solution of (5.41), let $s = U(0)$ and consider the curve $A = A(s)$. Then $A(s)$ has a unique (minimum) critical point at $s = s_c$, $A = A_c$. Moreover, define

$$\Phi(z) = \frac{\partial U(z;s)}{\partial s}\Big|_{s=s_c}.$$

Then $\Phi(z) > 0$ for all $z \geq 0$ and $\Phi \to 0$ as $z \to \infty$. Numerically, $A_c = -1.46638$, $s_c = -0.61512$.

The bifurcation diagram of A vs. $U(0)$ and some steady states is given by Fig. 5.3.

In particular, the profile U^- describes the shape of the finger within the boundary layer at the center of the droplet, which is responsible for the initiation of the splitting process. Similarly as was shown in [KWW07], the linearized

57

problem at the fold point has a zero eigenvalue; the corresponding eigenfunction is given by $\phi = \partial u/\partial\left[u(0)\right]$. Moreover, $\int \phi = 0$ due to mass conservation. As explained in [KWW07], it follows from Lemma 5.1.3 that ϕ has precisely one positive root; its profile is shown in Fig. 5.4. This proves that criterium 2 of Nishiura-Ueyema conditions is satisfied. This concludes the derivation of our result.

We use two methods to verify the droplet breakup condition from Result 5.1.2. We take $\varepsilon = 0.01$ and we let V_0 to be a slowly varying parameter in time, according to the formula $V_0 = 0.001t$. Using the initial condition

$$u(x,0) = \frac{1}{2}\left(1 - \tanh\left(\frac{|x| - 0.4}{0.01\sqrt{2}}\right)\right) \qquad (5.44)$$

we then compute numerically the solution to the full system (4.13). Droplet breakup is observed at about $t = 1982$ or $V_0 \approx 1.982$ as shown in Fig. 5.1. The initial conditions (5.44) correspond to $M = 0.8$. The formula (5.18) for V_c then yields $V_c = \frac{1.32606}{0.8^2} = 2.07$, which compares favorably with the numerical result.

Next, we computed the bifurcation diagram of the steady state (5.19, 5.20); this is shown in Fig. 5.2. To compute such diagram, we gradually changed $u(0)$ from 1 down to 0.5; then for each given $u(0)$, we used Maple's numerical boundary value problem solver to compute for the corresponding value of V_0. In this way, the fold point was found at $u(0) = 0.79$, with the corresponding $V_0 = 2.18$. This agrees very well with the asymptotic result $V_c = 2.07$ as well as (5.30) $u_m = 0.7887$.

5.2   Property of the advective Allen-Cahn equation

In this section we prove the existence, uniqueness and maximum principles for the advective Allen-Cahn equation and the advective nonlocal Allen-Cahn equation.

The maximum principle shows that the droplet breakup will not appear in many cases.

5.2.1 Existence and uniqueness

The existence and uniqueness for the advective Allen-Cahn equation can be done similarly to that of the advective Cahn-Hilliard equation. However, a different method has to be used for the advective nonlocal Allen-Cahn equation with mass conservation due to the extra nonlocal term. A semigroup method is used to show finite time existence of the solution, then a maximum principle analysis gives the bound of the λ in the non-local term.

Theorem 5.2.1 *For dimension $n = 1, 2, 3$, if $g(u)$ in (4.4) is a polynomial of order $2p$. then (4.15) and (4.16) with initial value $u_0 \in W^{\frac{3}{2},2}(\Omega)$ has a unique solution $u \in C^1([0,T]; C^2(\Omega))$, $\forall T > 0$.*

The proof contains two parts. The first part follows a similar process that is used in [Bal77, Hen81], which involves using the following propositions from them.

Proposition 5.2.2 *Consider the equation*

$$u_t = Au + N(u) \tag{5.45}$$

where A is the generator of a holomorphic semigroup $S(t)$ of bounded linear operators on a Banach space X. Suppose that $||S(t)|| \leq M_0$ for some constant $M_0 > 0$ for all $t > 0$. Under these hypotheses the fractional powers $(-A)^{-\alpha}$ can be defined for $0 \leq \alpha < 1$ and $(-A)^\alpha$ is a closed linear operator with domain $X_\alpha =$Domain$((-A)^\alpha)$ dense in X. Let $N(u)$ be locally Lipschitz, i.e. for each

bounded subset of U there exists a constant C_U such that

$$||N(u_1) - N(u_2)|| \leq C_U ||u_1 - u_2||_\alpha, \forall u, v \in U, \qquad (5.46)$$

then given $u_0 \in X$, there exists a finite time interval $[0, t)$ and a unique solution u with $u(0) = u_0$ on the time interval and the solution can be continued uniquely on a maximal interval of existence $[0, T^*)$. Moreover, if $T^* < \infty$ then $\lim_{t \to T^*} ||u(t)||_\alpha = \infty$.

Proposition 5.2.3 *Assume A and N same as above, suppose u is a solution of the equation on $(0, T]$, then if $\gamma < 1$, $t \to u_t(t) \in X_\gamma$ is locally Holder continuous for $t \in (0, T]$, with $||u_t||_\alpha \leq Ct^{\alpha - \gamma - 1}$.*

Lemma 5.2.4 *Assume A and N same as above, if $||N(u)|| \leq C(t)(1 + ||u||_\alpha)$, then the unique solution exists for all times.*

In (4.16), we can take $A = \epsilon\Delta$ on domain of $H^2(\Omega)$ functions with Neumann boundary condition, $1 > \alpha > \frac{3}{4}$, $X = L^2(\Omega)$ and

$$N(u) = -\nabla \cdot (u\vec{V}) - \frac{1}{\epsilon}f(u) + \lambda u. \qquad (5.47)$$

We have $X_\alpha \supset W^{\frac{3}{2}, 2}(\Omega) \cap L^\infty(\Omega)$. Thus, we can estimate the three terms of $N(u_1) - N(u_2)$ individually.

$$||\nabla \cdot (u_1\vec{V}) - \nabla \cdot (u_2\vec{V})||_{L^2} \leq ||\vec{V}||_{L^\infty}||\nabla u_1 - \nabla u_2||_{L^2} + ||\nabla \cdot \vec{V}||_{L^\infty}||u_1 - u_2||_{L^2}$$

$$\leq C||u_1 - u_2||_{H^1}$$

$$\leq C||u_1 - u_2||_{X^\alpha}.$$

$$(5.48)$$

Since f is a polynomial of order $2p - 1$ we have

$$f(u_1) - f(u_2) = (u_1 - u_2)h(u_1, u_2), \qquad (5.49)$$

where h is a polynomial of order $2p - 2$.

$$||f(u_1) - f(u_2)||_{L^2} \leq ||u_1 - u_2||_{L^2}||h(u_1, u_2)||_{L^\infty}$$

$$\leq C||u_1 - u_2||_{L^2}(||u_1||_{L^\infty}^{2p-2} + ||u_2||_{L^\infty}^{2p-2}) \qquad (5.50)$$

$$\leq C||u_1 - u_2||_{X^\alpha}(||u_1||_{X^\alpha}^{2p-1} + ||u_2||_{X^\alpha}^{2p-1}),$$

and

$$||u_1 \int_\Omega f(u_1) - u_2 \int_\Omega f(u_2)||_{L^2}$$

$$\leq ||u_1||_{L^2}||f(u_1) - f(u_2)||_{L^1} + ||u_1 - u_2||_{L^2}||f(u_2)||_{L^1}$$

$$\leq ||u_1 - u_2||_{L^1}||u_1||_{L^2}||h(u_1, u_2)||_{L^\infty} + C||u_1 - u_2||_{L^2}||u_2||_{L^{2p-1}}^{2p-1}$$

$$\leq C||u_1 - u_2||_{X^\alpha}||u_1||_{L^\infty}(||u_1||_{L^\infty}^{2p-2} + ||u_2||_{L^\infty}^{2p-2}) + C||u_1 - u_2||_{X^\alpha}||u_2||_{L^\infty}^{2p-1}$$

$$\leq C||u_1 - u_2||_{X^\alpha}||(||u_1||_{X^\alpha}^{2p-1} + ||u_2||_{X^\alpha}^{2p-1}).$$

$$(5.51)$$

We can apply proposition 5.2.2 from here and get a unique solution in $u \in D(A)$. Then, since $\nabla u \in W^{1,2}(\Omega) \subset L^6(\Omega)$, we have $Au = N(u) - \frac{du}{dt} \in L^6(\Omega)$. This implies $\nabla u \in W^{1,6}(\Omega)$, which is Holder continuous. This in turn shows $u \in C^{2+\delta}(\Omega)$ for some $\delta > 0$. The local Lipschitz condition for (4.15) is similar.

This proposition shows a maximum interval of existence $[0, T_{\max})$ for the advective Allen-Cahn equations. To show global existence, we need maximum principle below to show a bound for λ and $f(u)$, then we can directly apply lemma 5.2.4 using the fact of

$$||\nabla \cdot (u\vec{V})||_{L^2} \leq C||u||_{H^1} \leq C||u||_{X^\alpha}. \qquad (5.52)$$

5.2.2 Maximum principle analysis

The maximum principle-like analysis works only for the advective Allen-Cahn equation, since it is second-order and parabolic. The advective Cahn-Hilliard

equation, on the other hand, is of fourth-order thus does not possess the maximum principle.

Theorem 5.2.5 *For equation (4.15) with any velocity field, or (4.16) with expanding flow $\nabla \cdot \vec{V} \geq 0$, there exists a value u_M such that, if initial value $u_0(x) \in [0, u_M]$ in Ω and satisfies the condition of theorem 5.2.1, then $u(x,t) \in [0, u_M]$ for all t. For (4.16) with a general flow, $0 \leq u(x,t) \leq \max(\exp(-\inf(\nabla \cdot \vec{V})t), 1)u_M$.*

If we set $\hat{u}(x,t) = \exp^{\xi t} u(x,t)$, then (4.15) becomes

$$\hat{u}_t = \exp^{\xi t}(\epsilon \Delta u - \nabla u \cdot \vec{V} - u\nabla \cdot \vec{V} - \frac{1}{\epsilon}f(u) + \xi u), \qquad (5.53)$$

and (4.16) becomes

$$\hat{u}_t = \exp^{\xi t}(\epsilon \Delta u - \nabla u \cdot \vec{V} - u\nabla \cdot \vec{V} - \frac{1}{\epsilon}f(u) - \lambda u + \xi u). \qquad (5.54)$$

For the advective Allen-Cahn equation (4.15), we can simply take ξ close to 0. Since $g(u)$ is a double-well potential, $f(u) < 0$ when $u < 0$. We can deduce that there is no interior negative minima. Similarly, there is no interior maxima larger than 1. Thus, if the initial value is within [0,1], so is the solution.

For the advective nonlocal Allen-Cahn equation (4.16), it becomes a little complicated. Within any time interval [0,T], λ is bounded, so we can take a proper ξ in (5.54) to use the maximum principle. Thus, $\hat{u}$ has no negative minima within any interval (0,T]. If initial value is nonnegative, so is the solution.

The positive side is more tricky. If u takes its maximum value $u_{\max}$ on the interior and $\nabla \cdot \vec{V} \geq 0$, then on the point we have $\frac{1}{\epsilon}f(u_{\max}) + \lambda u_{\max} < 0$. On the other hand, due to the definition of λ (4.17), we know that $\lambda = -\frac{1}{\epsilon}\frac{f(u)}{u}$ for some $u \in [0, u_{\max}]$. This means that $\frac{f(u_{\max})}{u_{\max}} < \frac{f(u)}{u}$ for some $u \in [0, u_{\max}]$. Since $\frac{f(u)}{u}$

is even-ordered polynomial, there exists an $u_M > 0$ so that $\frac{f(u_{\max})}{u_{\max}} \geq \frac{f(u)}{u}$ for all $u_{\max} \geq u_M$ and $u \leq u_{\max}$. Thus, if the initial value is smaller than u_M, so is the solution. For example, if we take double-well potential $g(u) = u^2(1-u)^2$, then $u_M = 1.5$. For a general flow, we take $\xi = \inf(\nabla \cdot \vec{V})$ in (5.54). With a similar analysis, $u \leq u_M$ when $\hat{u}$ takes its maximum, hence the result.

For the advective Cahn-Hilliard equation (4.13), the maximum principle analysis does not work. In fact, there are cases when it fails: the solution becomes negative even when the initial value is not. See numerical results Fig. 6.3, Fig. 6.11 and Fig. 6.13.

In the simple 1D case, we can show the following fact:

Theorem 5.2.6 *If u satisfies (4.15) or (4.16), $u(x,0) \geq 0$ and bounded, and $u_x(x,0) \leq 0$ on Ω, and $V_{xx}(x) \geq 0$, then $u_x(x,t) \leq 0$ for all t.*

Note that, if we expect a symmetric condition, i.e. V is odd and u is even, and $u(x,0)$ takes its only maximum value at $x = 0$, then $u_x(0,t) = 0$, and we can apply this theorem on $\Omega \cap [0, \infty)$. Thus for any t, $u(x,t)$ takes the maximum value at $x = 0$, and the droplet breakup does not occur.

To prove this, we see that (4.15) leads to

$$(u_x)_t = (u_x)_{xx} - V(u_x)_x - (2V_x + f'(u))u_x - V_{xx}u, \qquad (5.55)$$

and (4.16) leads to

$$(u_x)_t = (u_x)_{xx} - V(u_x)_x - (2V_x + f'(u) + \lambda)u_x - V_{xx}u. \qquad (5.56)$$

Since $2V_x + f'(u)$ and λ are bounded, $V_{xx}u \geq 0$, we can use a process similar as above to show that no positive maximum can be achieved in the interior of Ω. Thus $u_x(x,t) \leq 0$ for all t. When V_{xx} is not nonnegative, breakup may occur. See Fig. 6.9 and Fig. 6.10.

In higher dimensions, it is easy to consider the case of radially symmetric data. General results require a more detailed analysis, but this analysis is suffice to show that Allen-Cahn type equation is unsuitable for the model of droplet breakup.

Theorem 5.2.7 *If u satisfies (4.15) or (4.16) on a n-dimensional sphere around 0, $u(\vec{x}, 0) \geq 0$, bounded and radially symmetric, and $u_r(\vec{x}, 0) \leq 0$ on Ω, where u_r is the directional derivative of u in the direction of $\vec{x}$. Assume $\vec{V}(\vec{x}) = V(|\vec{x}|)\frac{\vec{x}}{|\vec{x}|}$ and $r^2 V_{rr}(r) + (n-1)rV_r(r) - (n-1)V(r) \geq 0$ for any r, then $u_r(\vec{x}, t) \leq 0$ for all t.*

We can prove this by taking $w = r^{n-1}u_r$, then (4.15) gives

$$w_t = w_{rr} - (\frac{n-1}{r} - V)w_r - (2V_r + f'(u))w - r^{n-3}(r^2 V_{rr} + (n-1)rV_r - (n-1)V)u. \tag{5.57}$$

Using the same method as that of 1D case, we can show that no positive maxima exist under given condition. Thus, $w \geq 0$ for all $\vec{x}$ and t, which is equivalent as $u_r \geq 0$. When $n = 1$, the condition on $\vec{V}$ would be the same as the 1D theorem 5.2.6.

CHAPTER 6

Numerical Simulation for the Model Problem

6.1 Numerical Algorithm

In this section we present numerical simulation in 1,2 and 3D. We compare some of the results with the theory from previous sections. Specifically, we focus on different behaviors when the strength of velocity field changes, and different droplet breakup condition for different models. All our numerical results are consistent with the theories in previous sections.

The Cahn-Hilliard equation poses numerical challenges due to the stiffness of both the 4th-order term and the nonlinear term. Thus, many algorithms, both linear and nonlinear, have been proposed to solve it, for example finite element method [BBG98], and semi-implicit discretization [VR03, XT06, BJL10]. In this chapter we apply a simple semi-implicit splitting scheme [VR03] on the fourth-order term of the advective Cahn-Hilliard equation (4.13). It can be written as

$$\frac{u^{n+1} - u^n}{\Delta t} + \nabla \cdot (u^n \vec{V}) = -\Delta(\epsilon \Delta(A u^{n+1} + (1 - A) u^n) - \frac{1}{\epsilon} f(u^n)), \qquad (6.1)$$

where the advection term is discretized by the upwind scheme. The parameter A is chosen as 2 in the implementation. (4.15) and (4.16) are discretized as

$$\frac{u^{n+1} - u^n}{\Delta t} + \nabla \cdot (u^n \vec{V}) = \epsilon \Delta(A u^{n+1} + (1 - A) u^n) - \frac{1}{\epsilon} f(u^n) \qquad (6.2)$$

and

$$\frac{u^{n+1} - u^n}{\Delta t} + \nabla \cdot (u^n \vec{V}) = \epsilon \Delta (A u^{n+1} + (1 - A) u^n) - \frac{1}{\epsilon} f(u^n) + \lambda u \qquad (6.3)$$

respectively.

The stability condition now is related to $\vec{V}$. For example, the graph of stability of the advective Cahn-Hilliard equation related to time step Δt and the maximum norm of $\vec{V}$ is shown in Fig. 6.1. Δx has some insubstantial effects on the stability, but not so much as a CFL condition would require. In fact, the coefficient of u^{n+1} is $I + \Delta t \epsilon \Delta^2$, it is in the order of $(\Delta x)^{-4}$ when Δx is small. This is of a higher order than the advective term $\vec{V} \cdot \nabla u^n$, thus providing the main constraint for stability. This stability condition with $V = 0$ is consistent with similar results for the plain Cahn-Hilliard equations like [HLT07].Reference [BJL10] shows that a scheme of this kind would have an error of $O(C \Delta t)$, but the constant C would be very large. With the additional advection term, C becomes related to $V_{\max} = ||\vec{V}||_{L^\infty}$, thus when $V_{\max}$ increases, a smaller time step would be required. Moreover, when $\vec{V}$ is not very large, the most important constraint on Δt comes from the stability of the original Cahn-Hilliard equation.

6.2 Numerical Result

6.2.1 1D result: the advective Cahn-Hilliard equation

We begin from the basic 1D case where $u(x, 0) = \chi_{[-\alpha, \alpha]}$ and $\vec{V} = V_0 x$. The value of V_0 is tuned to show different types of solutions. The parameter ϵ is taken to be 0.01, and $g(u) = u^2(1 - u)^2$. α is taken as 0.3. We run the simulation on the interval $[-5, 5]$ with 2048 grid points. The time step is taken to be $\Delta t = 2 \times 10^{-6}$, with 5000 time steps in total. The result of the advective Cahn-Hilliard equation (4.13) contains two different types of solutions when $\vec{V}$ changes. When $\vec{V}$ is small,
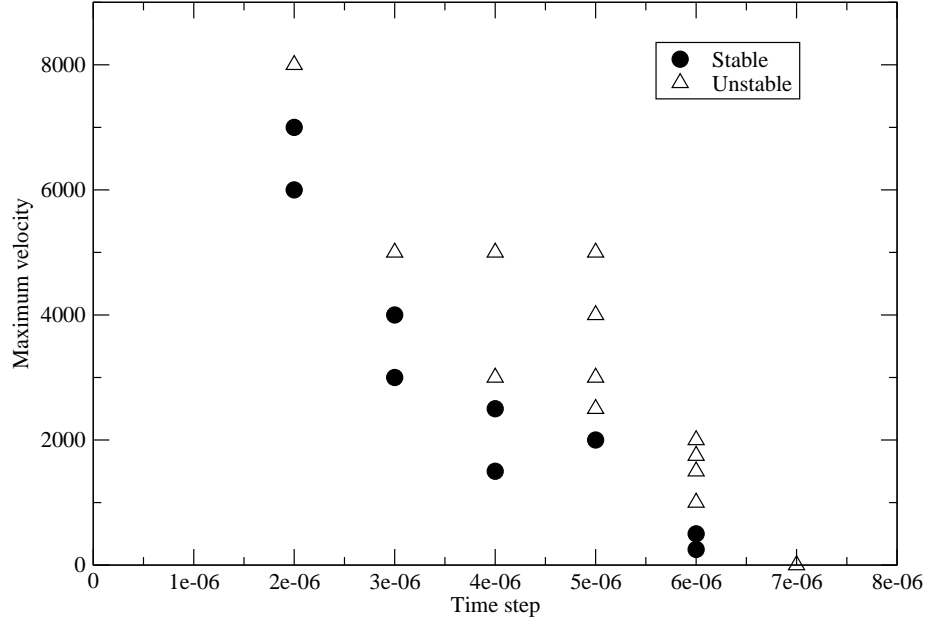
Figure 6.1: the Stability graph for the 1D advective Cahn-Hilliard equation. The triangle shape indicate unstable case, the circle represents the stable case. The X-axis and Y-axis represent the time step Δt and the maximum value of velocity field V_{max} respectively.

the solution develops a dimple in the middle, then stops, and does not break up further. When $\vec{V}$ is large, the solution eventually breaks up, and the smaller droplets continue to move apart. See Fig. 6.2 and Fig. 6.3, which correspond to $V_0 = 400$ and $V_0 = 600$ respectively. The threshold value of V_0 is drawn on Fig. 6.4, depending on the initial size of the droplet. The curve is an inverse quadratic curve of $V_0 M^2 = 1.326$, which fits the prediction of (5.18).
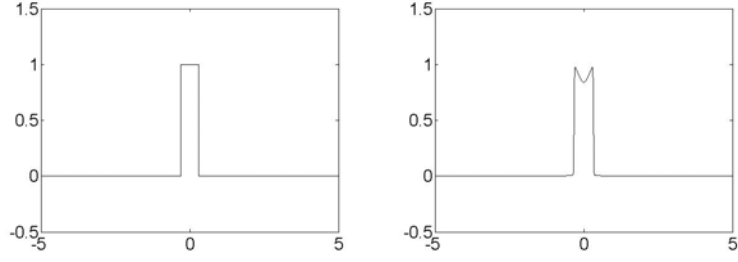
Figure 6.2: The advective Cahn-Hilliard equation does not breakup
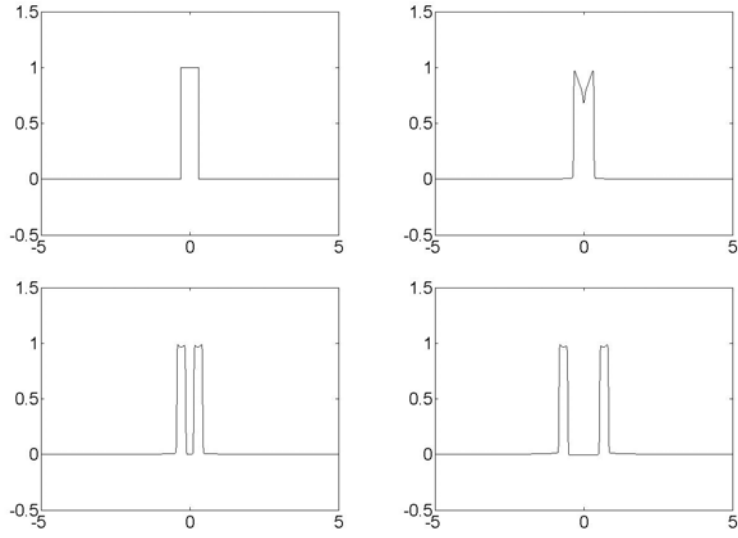
Figure 6.3: The advective Cahn-Hilliard equation breakup

6.2.2   1D result: the advective Allen-Cahn equation

As $\vec{V}$ increases, two different types of result appear for (4.15). When $\vec{V}$ is small, the solution develops towards a constant given by the solution of $V_0 u + \frac{1}{\epsilon} f(u) = 0$. When $\vec{V}$ is large and the above equation does not have a solution, the solution expands and decreases towards zero. The threshold is not related to α at all. See Fig. 6.5 and Fig. 6.6. Most numerical parameters are the same as that of the Cahn-Hilliard case: α is taken as 0.3, the simulation is on the interval $[-5, 5]$ with
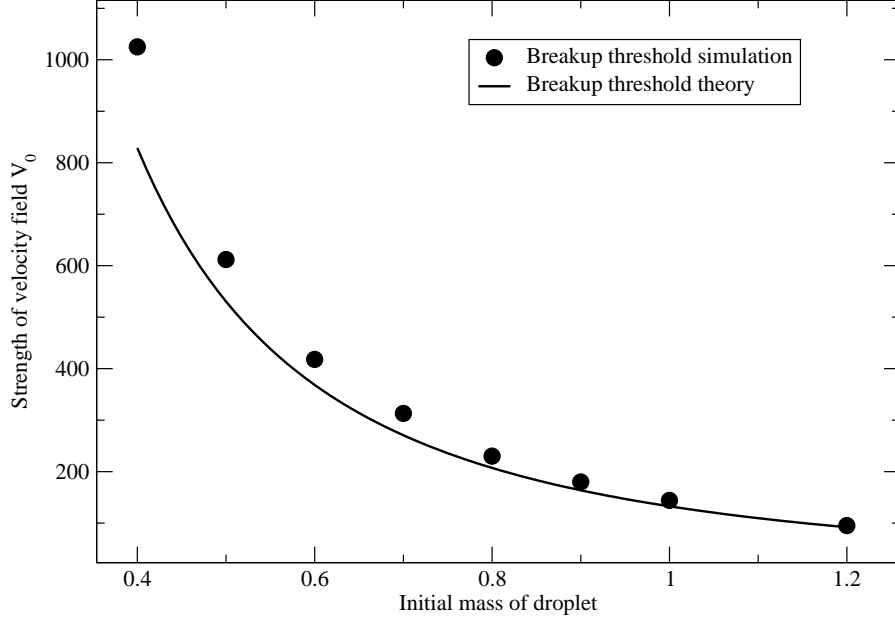
Figure 6.4: Threshold for Cahn-Hilliard. Dot is simulation data, line is an inverse quadratic curve $V_0 M^2 = 1.326$.

2048 grid points. The difference is in the time step and the strength of velocity field. In the graphs shown, the $\Delta t = 0.001$, and the values of V_0 are 10.0 and 30.0 respectively.

6.2.3 1D result: the advective nonlocal Allen-Cahn equation

Under the same setting, the advective nonlocal Allen-Cahn equation (4.16) has two different types of results when V_0 changes. The threshold value of V_0 is listed on Table 6.1. When Ω is smaller, these two thresholds also decrease. When $\vec{V}$ is small, the solution decreases and settles into a non-constant steady state

Figure 6.5: The advective Allen-Cahn equation when V is small

Figure 6.6: The advective Allen-Cahn equation when V is large

depicting a single droplet. When $\vec{V}$ is large, the solution decays to a small constant consistent with mass conservation. See Fig. 6.7 and Fig. 6.8. The

Table 6.1: Threshold for the advective nonlocal Allen-Cahn equation

Value of α	Threshold of V_0
0.5	7
0.2	18

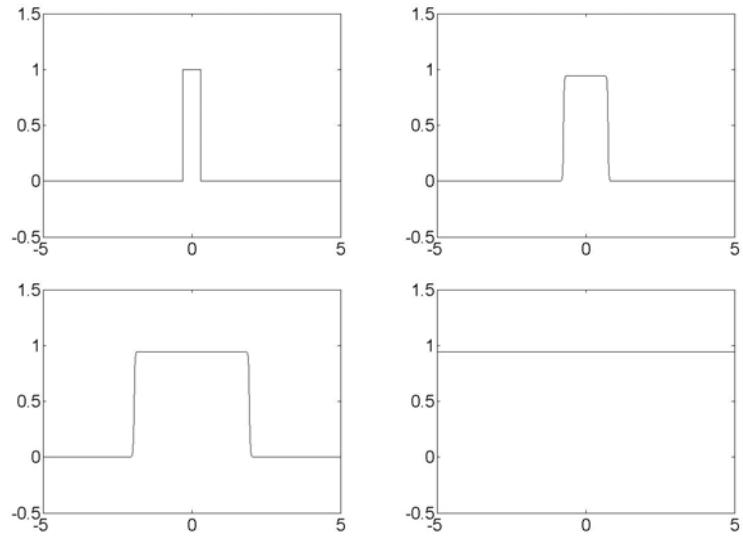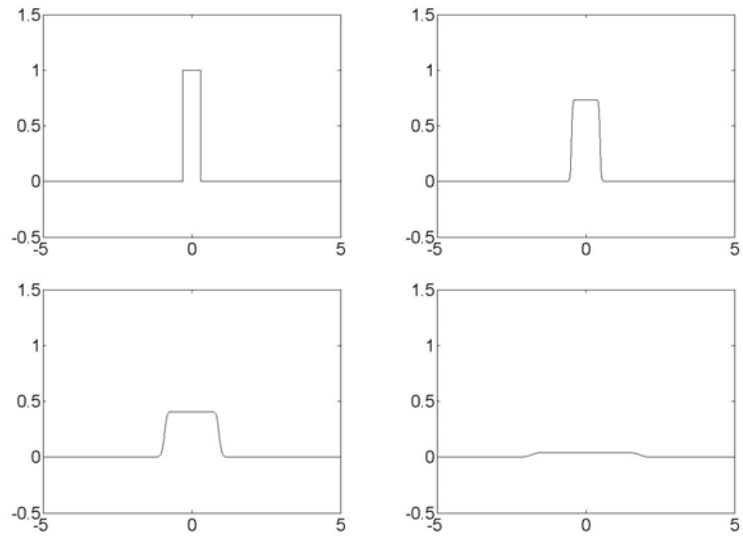numerical parameters are the same as in the previous subsection. α is taken as 0.3. The simulation is on interval $[-5, 5]$ with 2048 grid points. Time step $\Delta t = 0.001$, and the values of V_0 in the results shown are 10.0 and 30.0 respectively.
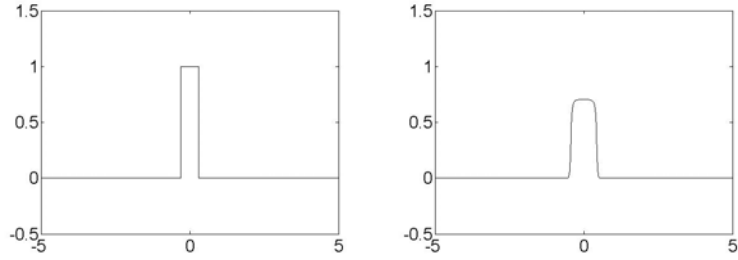
Figure 6.7: The advective nonlocal Allen-Cahn equation when V is small

This represents a typical Allen-Cahn solution that does not show droplet breakup. The reason comes from the maximum principle, which was explained in Theorem 5.2.5. However, if the initial value is non-monotone, things become different. Even a small concavity at the origin leads to a completely different evolution. In Fig. 6.9 we take $\vec{V}(x) = 5x$, but initial value is taken as 1 in $[-0.5, -0.01) \cup (0.01, 0.5]$, 0.99 in $[-0.01, 0.01]$, and 0 otherwise. The solution shows a breakup.

Another situation of droplet breakup involves a different velocity field $\vec{V}$. Fig. 6.10 is the result for the case when $V = V_0(x - \frac{1}{100}x^2)$ where $x \geq 0$ and expanded as an odd function to $x < 0$. Note that this velocity field does not satisfy the
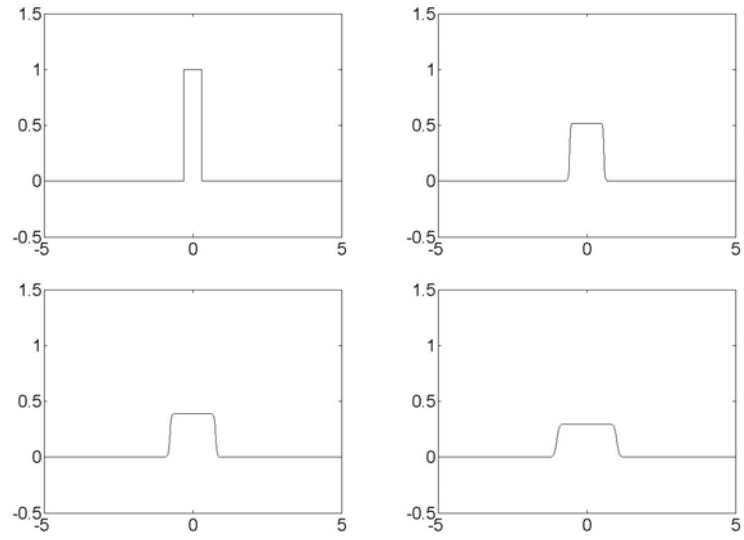
Figure 6.8: The advective nonlocal Allen-Cahn equation when V is large
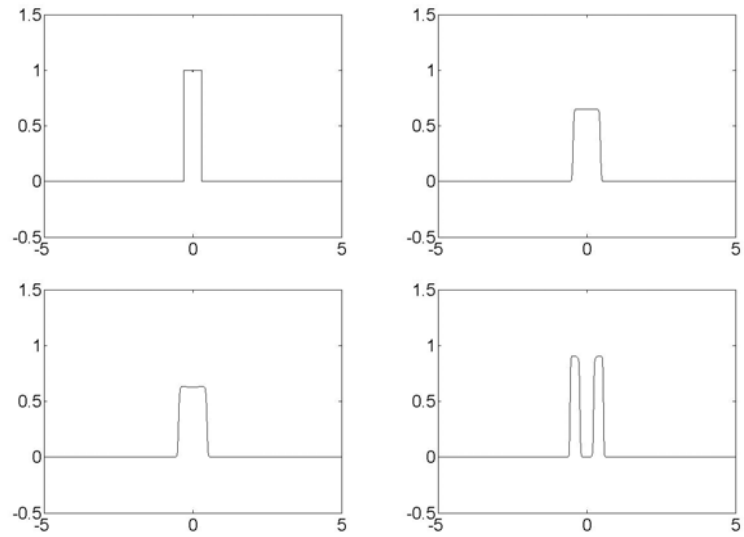
Figure 6.9: The advective nonlocal Allen-Cahn equation when the initial value have an insubstantial dent near the origin

condition of theorem 5.2.6. See Fig. 6.10.
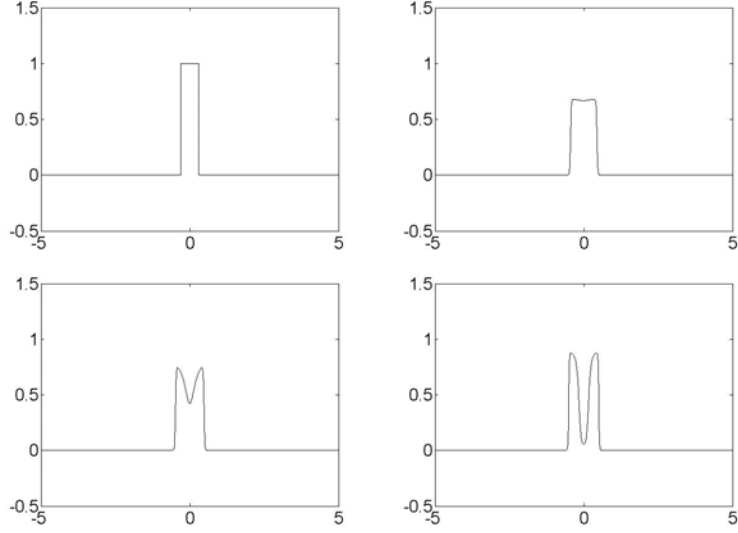
Figure 6.10: The advective nonlocal Allen-Cahn equation when V is not linear

6.2.4 2D result

Since the 1D case shows interesting results, it is natural to perform simulations in higher dimensions where we have additional geometry. We tried two different cases for 2D result, respectively under an expanding velocity field and a sheer flow. The velocity field is prescribed as

$$\vec{V}(x, y) = (V_0 x, V_0 y) \tag{6.4}$$

for the expanding case, and

$$\vec{V}(x, y) = (0, -V_0 x) \tag{6.5}$$

for the sheer flow. The advective Cahn-Hilliard equation and the advective nonlocal Allen-Cahn equation are both tested for these cases. For all cases, we solve

the equation in the region $[-1, 1] \times [-1, 1]$ with 128×128 mesh size. For the expanding flow, we test two cases with different initial values. The initial value for the first case is 1 on $[-0.3, 0.3] \times [-0.3, 0.3]$ and 0 otherwise. In the second case the initial value is 1 on a circle of radius 0.3 and 0 otherwise. For the sheer flow, the initial value is 1 on $[-0.1, 0.1] \times [-0.1, 0.1]$ and 0 otherwise. V_0 is 2000 for all the advective Cahn-Hilliard equation cases and 10 for all the advective nonlocal Allen-Cahn equation cases. Time step is 1×10^{-6} for the advective Cahn-Hilliard equation and 1×10^{-4} for the advective Allen-Cahn equation. These parameters are chosen to emphasize the difference in their breakup phenomena, see Fig. 6.11 to Fig. 6.16.

Similar to the 1D case, the advective Cahn-Hilliard equation has a droplet breakup, while the advective nonlocal Allen-Cahn equation does not. Comparatively, the Cahn-Hilliard model show a surface tension based breakup while Allen-Cahn model fails to do so in all cases.

6.2.5 3D result

For the 3D case, we used a parallel machine in the National Energy Research Scientific Computing Center (NERSC) to solve the problem. Due to the complexity of the problem, an operator splitting scheme is used. Instead of solving (6.1) directly, every time step is split into an advection step

$$\frac{u^* - u^n}{\Delta t} + \nabla \cdot (u^n \vec{V}) = 0, \tag{6.6}$$

and Cahn-Hilliard (or Allen-Cahn, respectively) step

$$\frac{u^{n+1} - u^*}{\Delta t} = -\nu \Delta (\epsilon \Delta (A u^{n+1} + (1 - A) u^*) - \frac{1}{\epsilon} f(u^*)). \tag{6.7}$$

The operator splitting and advection step are done by an ALE-AMR code [Ka10]. The Cahn-Hilliard step is solved by a specifically written finite element package.

Figure 6.11: The advective Cahn-Hilliard equation breakup under a 2D expanding flow with a square initial value

Figure 6.12: The advective nonlocal Allen-Cahn equation result under a 2D expanding flow with a square initial value

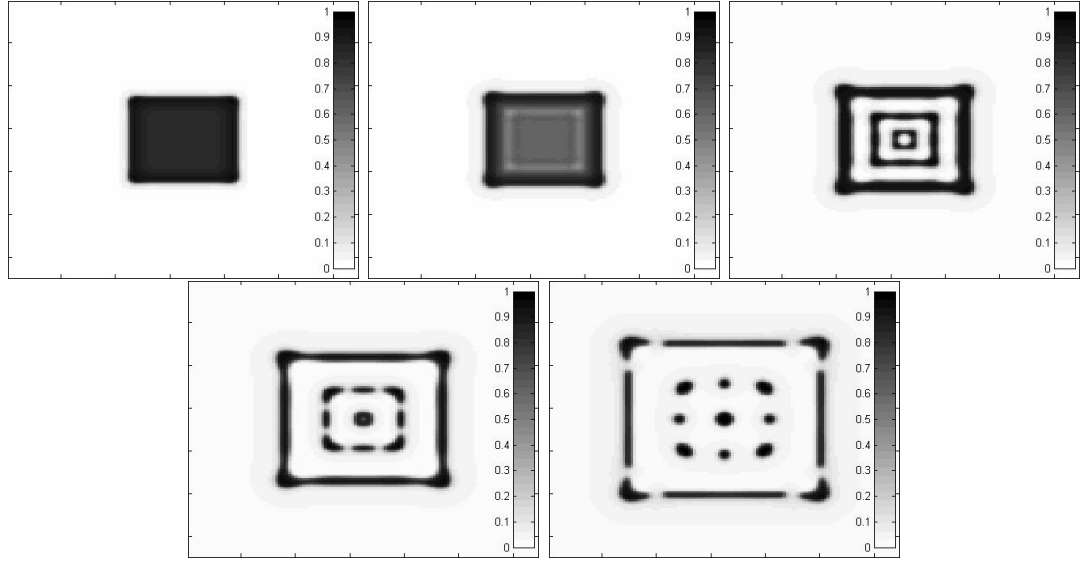Figure 6.13: The advective Cahn-Hilliard equation breakup under a 2D expanding flow with radially symmetric initial value

Figure 6.14: The advective nonlocal Allen-Cahn equation result under a 2D expanding flow with radially symmetric initial value

Figure 6.15: The advective Cahn-Hilliard equation breakup under a 2D sheer flow
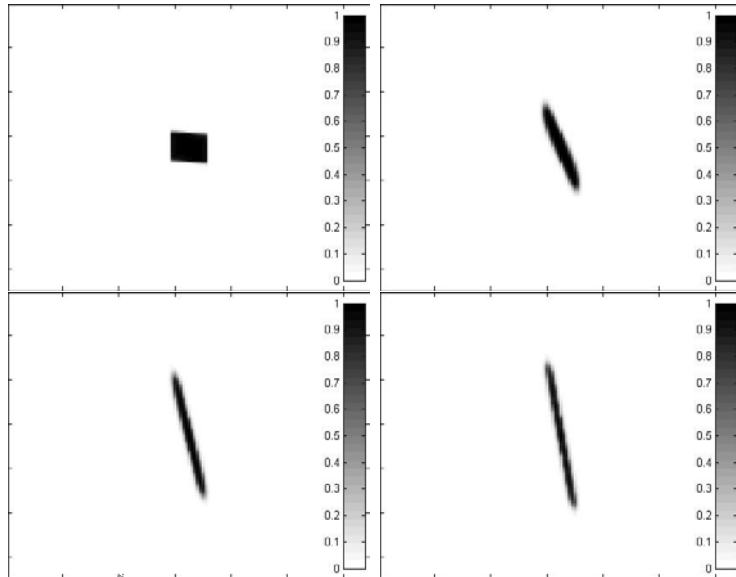
Figure 6.16: The advective nonlocal Allen-Cahn equation result under a 2D sheer flow

The simulation is run on a $[0, 1]^3$ grid, with initial value being 1 on $[0.35, 0.65]^3$ and 0 elsewhere. ϵ is still 0.01. The velocity field is prescribed as

$$\vec{V}(x, y, z) = (V_0(x - 0.5), V_0(y - 0.5), V_0(z - 0.5)). \tag{6.8}$$

where $V_0 = 1.0$. The time step is chosen adaptively by the ALE-AMR code. and the simulation shown is from time 0 to 1. The value of ν is 1×10^{-4} for all cases.

The advective Cahn-Hilliard have a droplet breakup similar to that of 2D case. The advective nonlocal Allen-Cahn equation simply performs a droplet expansion and then merge into the background or stop expanding, depending on the velocity field and droplet size. See Fig. 6.17 and Fig. 6.18.

6.2.6   Effect of Noise

The advective Allen-Cahn equation is more susceptible to noise compared to the advective Cahn-Hilliard equation. For the advective Allen-Cahn equation, even small noise in the initial value would lead to totally different behavior. However, the advective Cahn-Hilliard equation requires much stronger noise, or noise over time to make the result change. With strong enough noise, the droplet breakup shows some irregularity and breaks symmetry. Fig. 6.19 and 6.20 have the same setting as Fig. 6.11 and 6.12, except for a Gaussian noise of strength 0.01 added on the initial value. Fig. 6.21 and 6.22, on the other hand, adds a Gaussian noise every time step.

6.3   Concluding Remarks

In this part we focus on the properties and numerical simulation of the Cahn-Hilliard and Allen-Cahn equations with advection of a prescribed compressible flow. We have shown existence and uniqueness properties, and breakup condi-

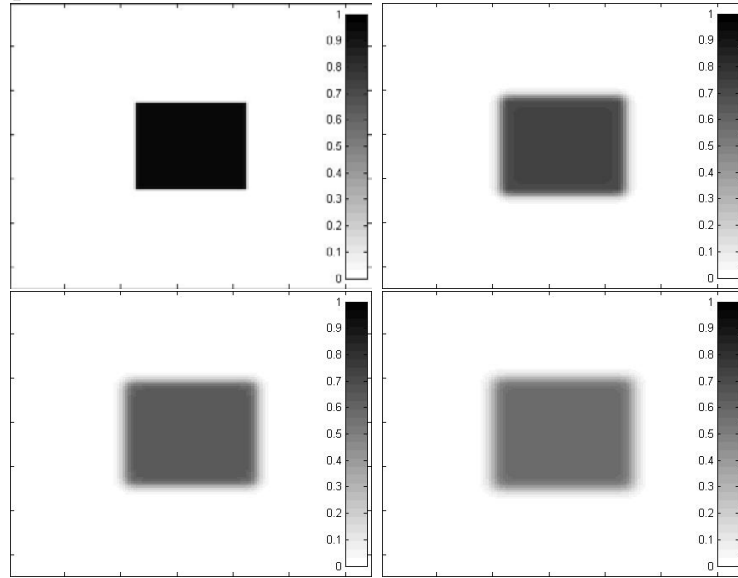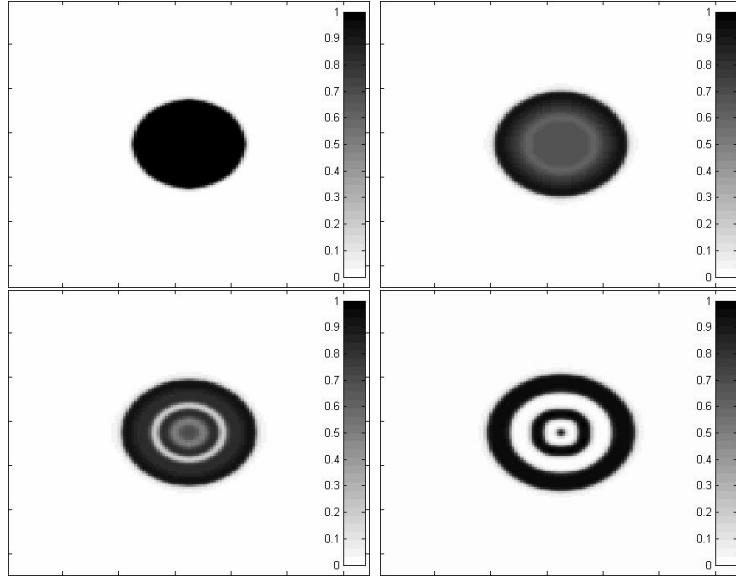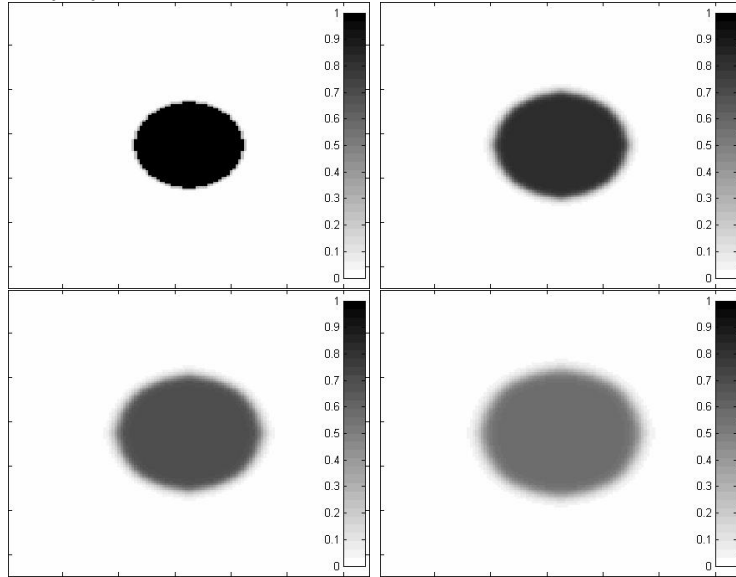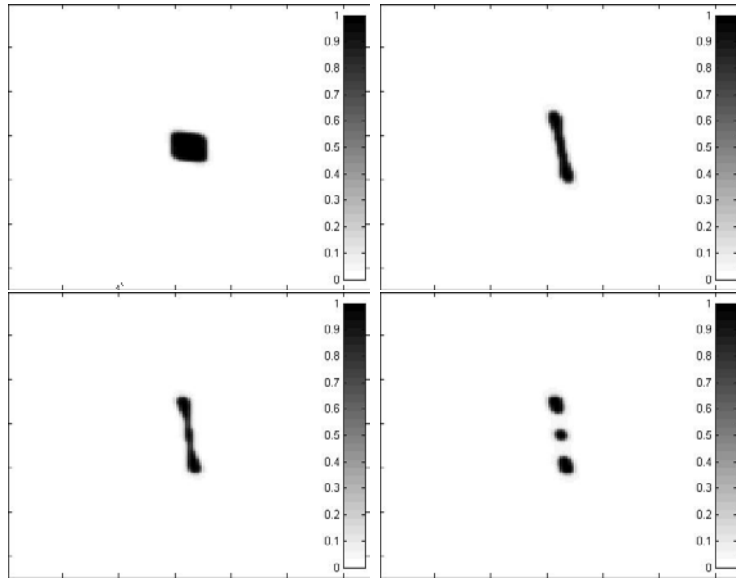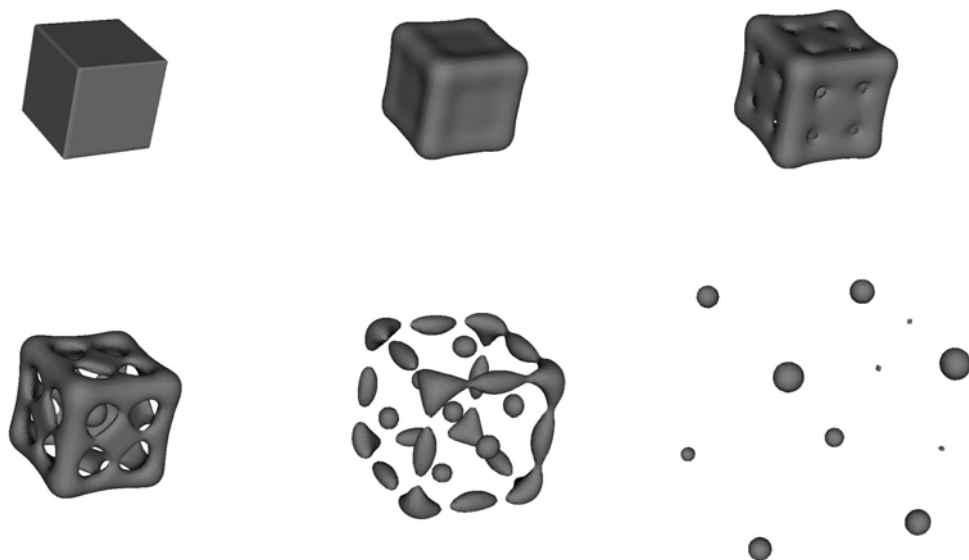Figure 6.17: The advective Cahn-Hilliard equation breakup under a 3D expanding flow
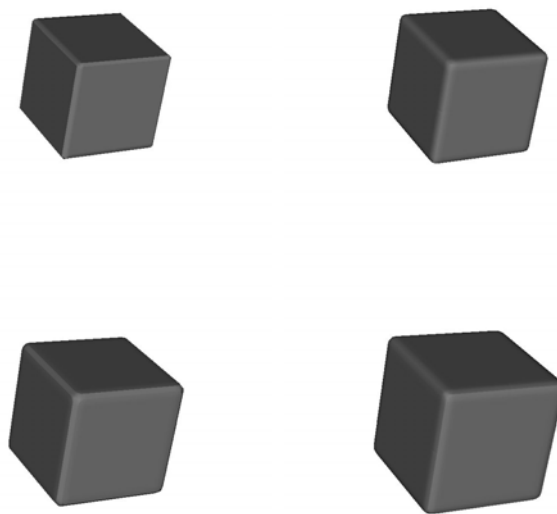
Figure 6.18: The advective nonlocal Allen-Cahn equation's result under a 3D expanding flow
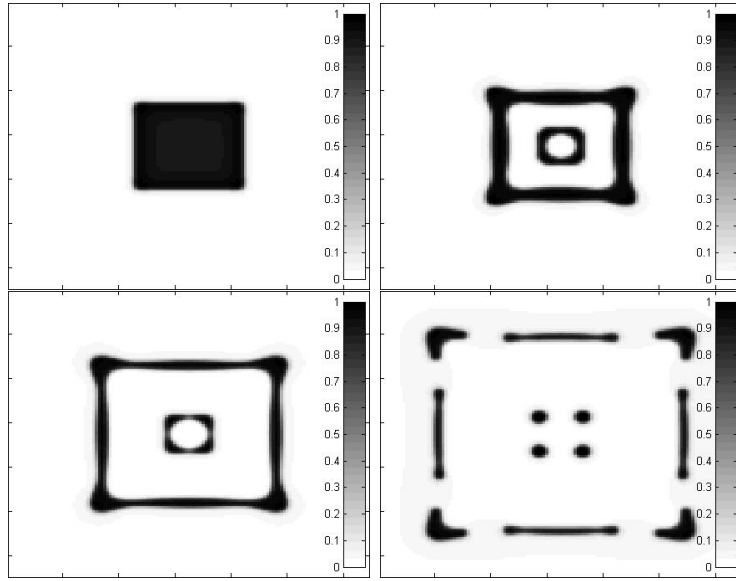
Figure 6.19: The advective Cahn-Hilliard equation breakup under a 2D expanding flow with noise of strength 0.01 in the initial value. It has a similar structure to that without noise.
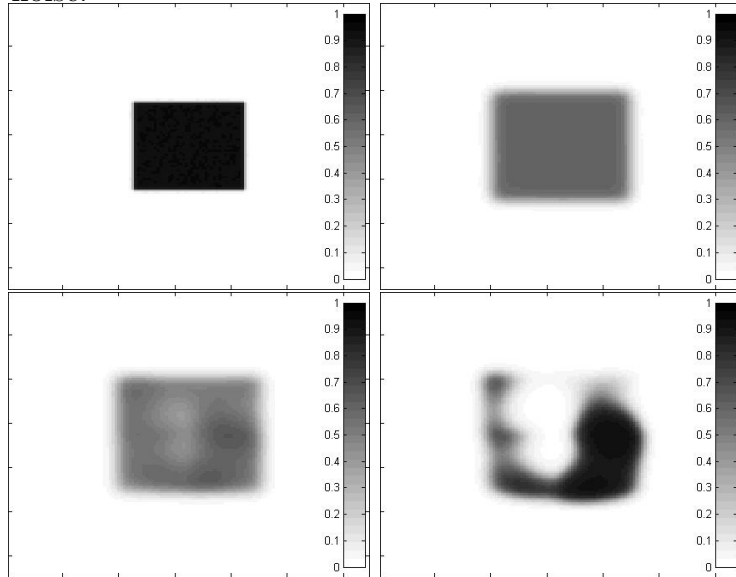
Figure 6.20: The advective nonlocal Allen-Cahn equation breakup under a 2D expanding flow with noise of strength 0.01 in the initial value. Without noise, it will not break up.
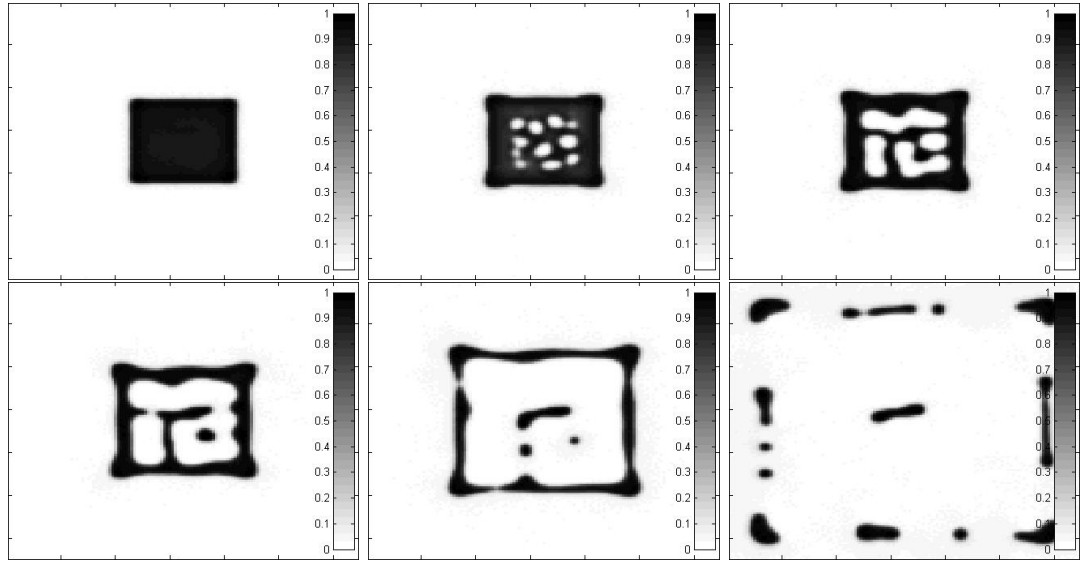
Figure 6.21: The advective Cahn-Hilliard equation breakup under a 2D expanding flow with continual noise over time. Symmetry is broken under this noise strength.
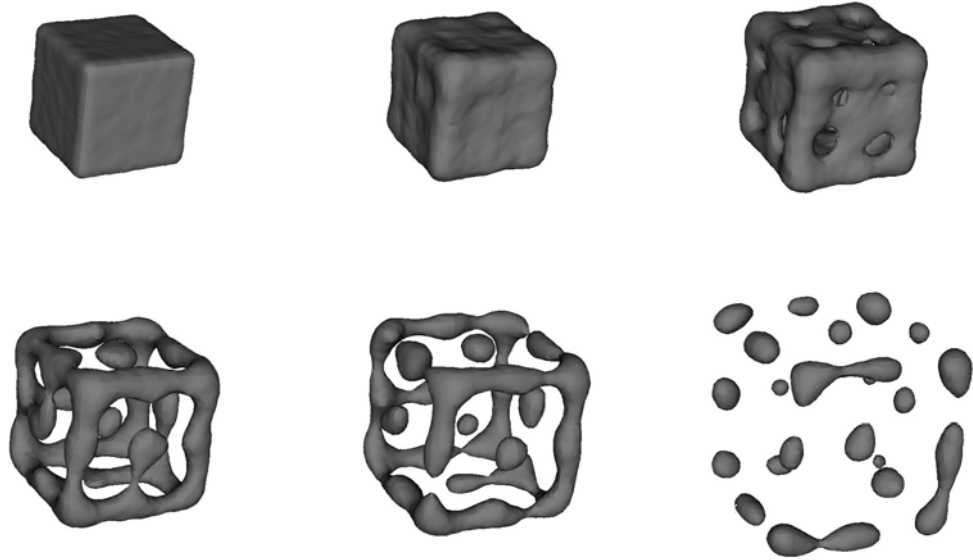
Figure 6.22: The advective Cahn-Hilliard breakup under a 3D expanding flow with continual noise over time. Symmetry is broken under this noise strength.

tions for both equations. For the advective Cahn-Hilliard equation, the droplet breakup condition is studied using a formal asymptotic analysis. It will happen when velocity field is large enough, and the threshold strength varies inverse quadratically with droplet size. For the advective Allen-Cahn equation, the breakup condition is studied using a maximum principle analysis. It will not happen without some kind of perturbation. Numerical results are provided in one, two and three space dimensions, with various initial conditions and different kinds of background flow. We also test numerical simulations with noise. The theoretical breakup condition fits well with the numerical condition.

Eventually we need to simulate the droplet breakup phenomenon with surface tension. For the future work, it is necessary to couple this model with other compressible fluid models. It is important to consider the impact of the phase field variable to the velocity field itself, and see how this model works within the full problem.

References

[ABC94]   Nicholas D. Alikakos, Peter W. Bates, and Xinfu Chen. "Convergence of the Cahn-Hilliard equation to the Hele-Shaw model." *Arch. Rational Mech. Anal.*, **128**(2):165–205, 1994.

[AF08]   Helmut Abels and Eduard Feireisl. "On a diffuse interface model for a two-phase flow of compressible viscous fluids." *Indiana Univ. Math. J.*, **57**(2):659–698, 2008.

[Bal77]   John Ball. "Remarks on the Blow-Up and nonexistence theorems for nonlinear evolution equations." *Quart. J. Math. Oxford*, **28**:473–486, 1977.

[BBG98]   John W. Barrett, James F. Blowey, and Harald Garcke. "Finite element approximation of a fourth order nonlinear degenerate parabolic equation." *Numer. Math.*, **80**(4):525–556, 1998.

[BBH09]   Shaunak D. Bopardikar, Francesco Bullo, and Joao P. Hespanha. "A cooperative Homicidal Chauffeur game." *Automatica*, **45**(7):1771 – 1777, 2009.

[BC05]   Francesco Bullo and Jorge Cortes. "Adaptive and Distributed Coordination Algorithms for Mobile Sensing Networks." In Vijay Kumar, Naomi Leonard, and A. Morse, editors, *Cooperative Control*, volume 309 of *Lecture Notes in Control and Information Sciences*, pp. 431–434. Springer, 2005.

[BJL10]   Andrea L. Bertozzi, N. Ju, and H. W. Lu. "A biharmonic modified forward time stepping method for fourth order nonlinear diffusion equations." to appear, DCDS, 2010.

[BJO96]   Ravi Bhagavatula, David Jasnow, and T. Ohta. "Nonequilibrium interface equations: An application to thermocapillary motion in binary systems." *Journal of Statistical Physics*, **88**:1013–1031, 9 1996.

[BLT09]   Martin Burger, Yanina Landa, Nicolay Tanushev, and Richard Tsai. "Discovering a Point Source in Unknown Environments." In Gregory Chirikjian, Howie Choset, Marco Morales, and Todd Murphey, editors, *Algorithmic Foundation of Robotics VIII*, volume 57 of *Springer Tracts in Advanced Robotics*, pp. 663–678. Springer, 2009.

[Boy99]   Franck Boyer. "Mathematical study of multi-phase flow under shear through order parameter formulation." *Asymptot. Anal.*, **20**(2):175–212, 1999.

[CFA01]   Rachel Caiden, Ronald P. Fedkiw, and Chris Anderson. "A numerical method for two-phase flow consisting of separate compressible and incompressible regions." *J. Comput. Phys.*, **166**(1):1–27, 2001.

[CH58]   John W. Cahn and John E. Hilliard. "Free Energy of a Nonuniform System. I. Interfacial Free Energy." *J. Chem. Phys.*, **28**:258–267, February 1958.

[CHD07]   Yao-Li Chuang, Y.R. Huang, M.R. D'Orsogna, and A.L. Bertozzi. "Multi-Vehicle Flocking: Scalability of Cooperative Control Algorithms using Pairwise Potentials." In *Robotics and Automation, 2007 IEEE International Conference on*, pp. 2292 –2299, 2007.

[CMT03]   B. Cook, D. Marthaler, C. Topaz, A. Bertozzi, , and M. Kemp. "Fractional bandwidth reacquisition algorithms for vsw-mcm." In *Multi-Robot Systems: From Swarms to Intelligent Automata*, volume 2, pp. 77–86. Kluwer Academic Publishers, 2003.

[COS09]   Jian-Feng Cai, Stanley Osher, and Zuowei Shen. "Linearized Bregman iterations for compressed sensing." *Mathematics of computation*, **78**, 2009.

[CS06]   R. Choksi and P. Sternberg. "Periodic phase separation: the periodic Cahn-Hilliard andisoperimetric problems." *Interfaces and Free Boundaries*, **8**:371–392, 2006.

[CWT09]   Alex Chen, Todd Wittman, Alexander Tartakovsky, and Andrea Bertozzi. "Image Segmentation Through Efficient Boundary Sampling." *UCLA CAM report*, 2009.

[DCB06]   M. R. D'Orsogna, Y. L. Chuang, A. L. Bertozzi, and L. S. Chayes. "Self-Propelled Particles with Soft-Core Interactions: Patterns, Stability, and Collapse." *Phys. Rev. Lett.*, **96**(10):104302, 2006.

[EFS05]   J. Enright, E. Frazzoli, K. Savla, and F. Bullo. "On Multiple UAV Routing with Stochastic Targets: Performance Bounds and Algorithms." In *the AIAA Conference on Guidance, Navigation, and Control, 2005*, 2005.

[Fa08]   A. Friedman and et al. "Toward a physics design for NDCX-II, an ion accelerator for warm dense matter and HIF target physics studies." *Lawrence Berkeley National Laboratory*, 2008.

[FPR10]   E. Feireisl, H. Petzeltova, E. Rocca, and G. Schimperna. "Analysis of a Phase-field Model for Two-phase Compressible Fluids." to appear in Math. Models Methods Appl. Sci, 2010.

[GCB08]   Chunkai Gao, Jorge Cortes, and Francesco Bullo. "Notes on averaging over acyclic digraphs and discrete coverage control." *Automatica*, **44**(8):2120 – 2127, 2008.

[Gra89]   Matthew A. Grayson. "A short note on the evolution of a surface by its mean curvature." *Duke Math. J.*, **58**(3):555–558, 1989.

[HCH06]   C.H. Hsieh, Yao-Li Chuang, Yuan Huang, K.K. Leung, A.L. Bertozzi, and E. Frazzoli. "An economical micro-car testbed for validation of cooperative control strategies." In *American Control Conference, 2006*, p. 6 pp., 2006.

[Hen81]   Daniel Henry. *Geometric theory of semilinear parabolic equations*, volume 840 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1981.

[HLT07]   Y. He, Y. Liu, and T. Tang. "On large time-stepping methods for the Cahn-Hilliard equation." *Applied Numerical Mathematics*, **57**(5-7):616 – 628, 2007. Special Issue for the International Conference on Scientific Computing.

[Jac99]   David Jacqmin. "Calculation of two-phase Navier-Stokes flows using phase-field modeling." *J. Comput. Phys.*, **155**(1):96–127, 1999.

[JB07]   Zhipu Jin and A.L. Bertozzi. "Environmental boundary tracking and estimation using multiple autonomous vehicles." In *Decision and Control, 2007 46th IEEE Conference on*, pp. 4918 –4923, 2007.

[JK04]   E. W. Justh and P. S. Krishnaprasad. "Equilibria and steering laws for planar formations." *Systems and Control Letters*, **52**(1):25–38, 2004.

[Ka10]   A. E. Koniges and et al. "ALE-AMR: A New 3D Multi-Physics Code for Modeling Laser/Target Effects." to appear in J. Physics, 2010.

[KKL04]   Junseok Kim, Kyungkeun Kang, and John Lowengrub. "Conserva-
           tive multigrid methods for Cahn-Hilliard fluids." *J. Comput. Phys.*,
           193(2):511–543, 2004.

[KMK07]   Allison Kolpas, Jeff Moehlis, and Ioannis G. Kevrekidis. "Coarse-
           grained analysis of stochasticity-induced switching between collective
           motion states." *Proceedings of the National Academy of Sciences*,
           104(14):5931–5935, 2007.

[KO94]   B. S. Kerner and V. V. Osipov. *Autosolitions: A New Approach
           to Problem of Self-Organization and Turbulence.* Kluwer Academic
           Publishers, Dordrecht, 1994.

[KS89]   Robert V. Kohn and Peter Sternberg. "Local minimisers and singular
           perturbations." *Proc. Roy. Soc. Edinburgh Sect. A*, **111**(1-2):69–84,
           1989.

[KWW07]   T. Kolokolnikov, M.J. Ward, and J. Wei. "Self-replication of mesa
           patterns in reaction-diffusion models." *Physica D: Nonlinear Phe-
           nomena*, **236**(2):104 – 122, 2007.

[LBK10]   Wangyi Liu, Andrea L. Bertozzi, and Theodore Kolokolnikov. "Dif-
           fuse interface surface tension models in an expanding flow." *Preprint*,
           2010.

[LGB07]   H.-W. Lu, K. Glasner, A. Bertozzi, and C.-J. Kim. "A Diffuse Inter-
           face Model for Electrowetting Droplets In a Hele-Shaw Cell." *J. Fluid
           Mechanics*, **590**:411–435, 2007.

[LHH07]   K.K. Leung, C.H. Hsieh, Y.R. Huang, A. Joshi, V. Voroninski, and
           A.L. Bertozzi. "A second generation micro-vehicle testbed for coop-
           erative control and sensing strategies." In *American Control Confer-
           ence, 2007. ACC '07*, pp. 1900 –1907, 2007.

[LSB09]   Wangyi Liu, Martin B. Short, Andrea L. Bertozzi, and Yasser E.
           Taima. "Multiscale collaborative searching through swarming." *Pro-
           ceedings of the 7th International Conference on Informatics in Con-
           trol, Automation and Robotics*, 2009.

[LST10]   Chun Liu, Norifumi Sato, and Yoshihiro Tonegawa. "On the existence
           of mean curvature flow with transport term." *Interfaces and Free
           Boundaries*, **12**(2), 2010.

[LT98]   J. Lowengrub and L. Truskinovsky. "Quasi-incompressible Cahn-Hilliard fluids and topological transitions." *R. Soc. Lond. Proc. Ser. A Math. Phys. Eng. Sci.*, **454**(1978):2617–2654, 1998.

[NCT05]   B.Q. Nguyen, Yao-Ling Chuang, D. Tung, Chung Hsieh, Zhipu Jin, Ling Shi, D. Marthaler, A. Bertozzi, and R.M. Murray. "Virtual attractive-repulsive potentials for cooperative control of second order dynamic vehicles on the Caltech MVWT." In *American Control Conference, 2005. Proceedings of the 2005*, pp. 1084 – 1089 vol. 2, 2005.

[NU99]   Y. Nishiura and D. Ueyama. "A Skeleton Structure of Self-Replicating Dynamics." *Physica D*, **130**(1):73 – 104, 1999.

[Olf07]   R. Olfati-Saber. "Distributed Tracking for Mobile Sensor Networks with Information-Driven Mobility." In *American Control Conference, 2007. ACC '07*, pp. 4606 –4612, 2007.

[PAG54]   E. S. PAGE. "CONTINUOUS INSPECTION SCHEMES." *Biometrika*, **41**(1-2):100–115, 1954.

[Peg89]   R. L. Pego. "Front migration in the nonlinear Cahn-Hilliard equation." *Proc. Roy. Soc. London Ser. A*, **422**(1863):261–278, 1989.

[PPD94]   Sanjay Puri, Nita Parekh, and Sushanta Dattagupta. "Phase ordering dynamics in a gravitational field." *Journal of Statistical Physics*, **75**:839–857, 6 1994.

[RS92]   J. Rubinstein and P. Sternberg. "Nonlocal reaction-diffusion equations and nucleation." *J. Appl. Math.*, **48**(3):249 – 264, 1992.

[SB09]   Stephen L. Smith and Francesco Bullo. "The dynamic team forming problem: Throughput and delay for unbiased policies." *Systems and Control Letters*, **58**(10-11):709 – 715, 2009.

[SO09]   Mark Sussman and Mitsuhiro Ohta. "A stable and efficient method for treating surface tension in incompressible two-phase flow." *SIAM J. Sci. Comput.*, **31**(4):2447–2471, 2009.

[SPL08]   R. Sepulchre, D.A. Paley, and N.E. Leonard. "Stabilization of Planar Collective Motion With Limited Communication." *Automatic Control, IEEE Transactions on*, **53**(3):706 –719, 2008.

[SSH07]   M. Sussman, K. M. Smith, M. Y. Hussaini, M. Ohta, and R. Zhi-Wei. "A sharp interface method for incompressible two-phase flows." *J. Comput. Phys.*, **221**(2):469–505, 2007.

[SSO94]   Mark Sussman, Peter Smereka, and Stanley Osher. "A level set approach for computing solutions to incompressible two-phase flow." *J. Comput. Phys.*, **114**(1):146–159, 1994.

[Tem88]   Roger Temam. *Infinite-dimensional dynamical systems in mechanics and physics*, volume 68 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 1988.

[Tem95]   Roger Temam. *Navier-Stokes equations and nonlinear functional analysis*, volume 66 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 1995.

[Tra07]   John Travis. "Do Wandering Albatrosses Care About Math?" *Science*, **318**(5851):742–743, 2007.

[TSO92]   M. Tjahjadi, H. A. Stone, and J. M. Ottino. "Satellite and subsatellite formation in capillary breakup." *Journal of Fluid Mechanics*, **243**(-1):297–317, 1992.

[VBH99]   G. M. Viswanathan, S. V. Buldyrev, S. Havlin, M. G. E. da Luz, E. P. Raposo, and H. E. Stanley. "Optimizing the success of random searches." *Nature*, **401**:911–914, October 1999.

[VCB95]   Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. "Novel Type of Phase Transition in a System of Self-Driven Particles." *Phys. Rev. Lett.*, **75**(6):1226–1229, Aug 1995.

[VR03]   Benjamin P. Vollmayr-Lee and Andrew D. Rutenberg. "Fast and accurate coarsening simulation with an unconditionally stable time step." *Phys. Rev. E*, **68**(6):066703, 2003.

[XT06]   Chuanju Xu and Tao Tang. "Stability analysis of large time-stepping methods for epitaxial growth models." *SIAM J. Numer. Anal.*, **44**(4):1759–1779, 2006.

[YFL05]   Pengtao Yue, James J. Feng, Chun Liu, and Jie Shen. "Diffuse-interface simulations of drop coalescence and retraction in viscoelastic fluids." *Journal of Non-Newtonian Fluid Mechanics*, **129**(3):163 – 176, 2005.